

# Secure Cooperative Regenerating Codes for Distributed Storage Systems

O. Ozan Koyluoglu, Ankit Singh Rawat, and Sriram Vishwanath

**Abstract**—Regenerating codes enable trading off repair bandwidth for storage in distributed storage systems (DSS). Due to their distributed nature, these systems are intrinsically susceptible to attacks, and they may be susceptible to multiple node failures. This paper analyzes storage systems that employ cooperative regenerating codes that are robust to (passive) eavesdroppers. The analysis is divided into two parts, studying both minimum bandwidth and minimum storage cooperative regenerating scenarios. First, the secrecy capacity of minimum bandwidth cooperative regenerating codes is characterized. Second, for minimum storage cooperative regenerating codes, a secure file size upper bound and achievability results are provided. These results establish the secrecy capacity for the minimum storage scenario for certain special cases. In all scenarios, the achievability results correspond to exact repair, and secure file size upper bounds are obtained using mincut analyses over a suitable secrecy graph representation of DSS. The main achievability argument is based on appropriate precoding of the data to eliminate the information at the eavesdropper.

**Index Terms**—Coding for distributed storage systems, cooperative repair, minimum bandwidth cooperative regenerating (MBCR) codes, minimum storage cooperative regenerating (MSCR) codes, security.

## I. INTRODUCTION

Distributed storage systems (DSS) are designed to store data over a distributed network of nodes. DSS have become increasingly important given the growing volumes of data being generated, analyzed and archived today. OceanStore [1], Google File System (GFS) [2], and TotalRecall [3] are a few examples of storage systems employed today. Data to be stored is more than doubling every two years, and efficiency in storage and data recovery is particularly critical today. The coding schemes employed by DSS are designed to provide efficient storage while ensuring resilience against node failures in order to prevent the permanent loss of the data stored on the system. In a majority of existing literature, the analysis of DSS focuses primarily on isolated node failures. In our work, we study a more general scenario of DSS that can suffer from multiple simultaneous node failures. In addition to multiple node failures, DSS systems are also inherently susceptible to adversarial attacks, such as one from eavesdroppers aiming to gain access to the stored data. Therefore, a “good” DSS would meet desired security requirements while performing efficient repairs even in the case of multiple simultaneous node failures.

In [4], Dimakis et al. present a class of *regenerating codes*, which efficiently trade-off per node storage and repair

bandwidth for single node repair. These codes are designed to possess a maximum distance separable (MDS) property, which is an “any  $k$  out of  $n$ ” property, wherein the entire data can be reconstructed by contacting to any  $k$  storage nodes out of  $n$  nodes. By utilizing a network coding approach, the notion of *functional repair* is developed in [4], where the original failed node may not be replicated exactly, but can be repaired such that it is *functionally* equivalent. On the other hand, *exact repair* requires that the regeneration process results in an exact replica of the data stored on the failed node. This is essential due to the ease of maintenance and other practical purposes such as maintaining a code in its systematic form. Exact repair may also prove to be advantageous compared to functional repair in the presence of eavesdroppers, as the latter scheme requires updating the coding coefficients, which in turn may leak additional information to eavesdroppers [5]. The design of exact regenerating codes achieving one of the two ends of the trade off between storage and repair bandwidth has been recently investigated by researchers. In particular, Rashmi et al. [6] design codes that are optimal for all parameters at the minimum bandwidth regeneration (MBR) point. For the minimum storage regeneration (MSR) point, optimal codes are presented in multiple recent papers. (See [7]–[10] and references therein.)

As discussed before, DSS can also exhibit multiple simultaneous node failures, and it is desirable that these be repaired simultaneously. It is not uncommon that multiple failures occur in DSS, especially for large-scale systems. In addition, some DSS administrators may choose to wait to initiate a repair process after a critical number of failures has occurred (say  $t$  of them), in order to render the entire process more efficient and less frequent. For example, TotalRecall [3] currently executes a node repair process only after a certain threshold on the number of failures is reached. In such multiple failure scenarios, each new node replacing a failed one can still contact  $d$  remaining (surviving) nodes to download data for the repair process. In addition, replacement nodes, after downloading data from surviving nodes, can also exchange data within themselves to complete the repair process. This repair process is referred to as *cooperative repair* in [11], which present network coding techniques to implement such repairs. Cooperative repair is shown to be essential as it can help in lowering the total repair bandwidth compared to the  $t = 1$  case. Flexibility of the choice on download nodes at repair nodes is analyzed in [12]. [13], focusing on functional repair, shows that under the constraint  $n = d + t$ , deliberately delaying repairs (and thus increasing  $t$ ) does not result in gains in terms of MBR/MSR optimality. [13] and [14] utilize a cut-

The authors are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712, USA. (e-mail: ozan@mail.utexas.edu, ankitsr@utexas.edu, sriram@ece.utexas.edu.)

set bound argument and derive the cooperative counterpart of the end points of the trade off region. These two points are named as the minimum bandwidth cooperative regenerating (MBCR) point and the minimum storage cooperative regenerating (MSCR) point (See also [15]). The work in [14] shows the existence of cooperative regenerating codes with optimal repair bandwidth. Explicit code constructions for exact repair on this setup are presented in [16], for the MBCR point, and in [17], for the MSCR point. These constructions are designed for the setting of  $d = k$ . (See also [18].) Interference alignment is used in [19] to construct scalar codes to operate at the MSCR point. (This construction is limited to the case  $k = 2$  with  $d \geq k$ , and does not generalize to  $k \geq 3$  with  $d > k$ .) An explicit construction for the MBCR point, with the restriction that  $n = d + t$  for any  $t \geq 1$ , is presented in [20]. Finally, the reference [21] presents designs of scalar codes for the MBCR point for all possible parameter values. Noting the significance of cooperative repair in DSS, regenerating codes that have resilience to eavesdropping attacks will have greater value if they also have efficient cooperative repair mechanisms.

The security of systems can be understood in terms of their resilience to either (or both) active or passive attacks [22], [23]. Active attacks include settings where the attacker modifies existing packets or injecting new ones to the system, whereas passive attacks include eavesdroppers observing the information being stored/transmitted. For DSS, cryptographic approaches like private-key cryptography are often logistically prohibitive, as the secret key distribution between each pair of nodes and its renewal are highly challenging, especially for large-scale systems. In addition, most cryptographic approaches are typically based on certain hardness results, which, if repudiated, could leave the system vulnerable to attacks. On the other hand, information theoretic security, see, e.g., [24], [25], presents secrecy guarantees even with infinite computational power at eavesdroppers without requiring the sharing and/or distribution of keys. This approach is based on the design of secrecy-achieving coding schemes by taking into account the amount of information leaked to eavesdroppers, and can offer new solutions to security challenges in DSS. In its simplest form, the security can be achieved with the one-time pad scheme [26], which claims the security of the ciphertext obtained by XOR of data and uniform key. This approach is of significant value to DSS. For example, consider a system storing the key at a node, and ciphertext at another node. Then, the eavesdropper will not obtain any information by observing one of these two nodes, whereas the data collector can contact to both nodes and decipher the data.

The problem of designing secure DSS against eavesdropping attacks has been recently studied by Pawar et al. [5], where the authors consider a passive eavesdropper model that observe the data stored on  $\ell$  ( $< k$ ) storage nodes for a DSS employing an MBR code. The proposed schemes are designed for the “bandwidth limited regime”, and shown to achieve an upper bound on the secure file size, establishing its optimality. Shah et al. [27] consider the design of secure MSR codes. Here, they show that the eavesdropper model for an MSR code should be extended compared to that of an MBR code. The underlying reason is that at the MSR point of

operation, the eavesdropper may obtain additional information by observing the downloaded information (as compared to just observing the stored information). Thus, at the MSR point, the eavesdropper is modeled with a pair  $(\ell_1, \ell_2)$  with  $\ell_1 + \ell_2 < k$ , where the eavesdropper has knowledge of the content of the  $\ell_1$  number of nodes, and, in addition, has knowledge of the downloaded information (and hence also the storage content) of the  $\ell_2$  number of nodes. We note that, as the downloaded data is stored for minimum bandwidth regenerating codes, the two notions are different only at the minimum storage point. Considering such an eavesdropper model, Shah et al. present coding schemes utilizing product matrix codes [6], and show that the bound on secrecy capacity in [5] at MBR is achievable. They further use product matrix based codes for MSR point as well, and show the bound in [5] is achievable only when  $\ell_2 = 0$ . In addition to this classical MBR/MSR setting, the security aspects of locally repairable codes (see, e.g., [28]–[33]) are studied in [34]; and security against active eavesdroppers are investigated in [35]–[37].

In this paper, we analyze and design secure and cooperative regenerating codes for DSS. In terms of security requirements, we utilize a passive and colluding eavesdropper model as presented in [27]. In this model, during the entire life span of the DSS, the eavesdropper can gain access to data stored on an  $\ell_1$  number of nodes, and, in addition, it observes both the stored content and the data downloaded (for repair) on an additional  $\ell_2$  number of nodes. Given this eavesdropper model, we focus on the problem of designing secure regenerating codes in the context of DSS that performs multiple node repairs in a cooperative manner. This scenario generalizes the single node repair setting considered in earlier works to multiple node failures. First, we present upper bound on the secrecy capacity for MBCR codes, and present a secure coding scheme that achieves this bound. This proves the tightness of the bound and characterizes the secrecy capacity for MBCR codes. Next, we address the secrecy capacity of a DSS employing the MSCR codes, and show that the existing MSCR codes can be made secure against eavesdropping. In this minimum storage setup, our codes match the upper bound secure file size under special cases. In all scenarios, the achievability results allows for exact repair, and secure file size upper bounds are obtained from mincut analyses over the secrecy graph representation of DSS. The main secrecy achievability coding argument of the paper is obtained by utilizing a secret precoding scheme to obtain secure coding schemes for DSS. In some cases, this precoding is established simply with the one-time pad scheme, and in others *maximum rank distance* (MRD) codes are utilized similar to the classical work of [38].

The rest of the paper is organized as follows. In Section II, we provide the general system model together with some preliminary results utilized throughout the text. Section III provides the analysis of secure MBCR codes, and Section IV is devoted to the secure MSCR codes. The paper is concluded in Section V, and, to enhance the flow of the paper, some of the results and proofs are relegated to appendices.

## II. SYSTEM MODEL AND PRELIMINARIES

Consider a DSS with  $n$  live nodes at a time and a file  $\mathbf{f}$  of size  $\mathcal{M}$  over  $\mathbb{F}_q$  that needs to be stored on the DSS. In order to store the file  $\mathbf{f}$ , it is divided into  $k$  blocks of size  $\frac{\mathcal{M}}{k}$  each. Let  $(\mathbf{f}_1, \dots, \mathbf{f}_k)$  denotes these  $k$  blocks. Here, we have  $\mathbf{f}_i \in \mathbb{F}_q^{\frac{\mathcal{M}}{k}}$ . These  $k$  data blocks are encoded into  $n$  data blocks,  $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ , each of length  $\alpha$  over  $\mathbb{F}_q$  ( $\alpha \geq \frac{\mathcal{M}}{k}$ ). Given the codewords, node  $i$  in an  $n$ -node DSS stores encoded block  $\mathbf{x}_i$ . In this paper, we use  $\mathbf{x}_i$ , to represent both block  $\mathbf{x}_i$  and a storage node storing this encoded block interchangeably. Motivated by the MDS property of the codes that are traditionally proposed to store data in centralized storage systems [39]–[41], the works on regenerating codes focus on storage schemes that have “any  $k$  out of  $n$ ” property, i.e., the content of any  $k$  nodes will suffice to recover the file. We focus on codes achieving this property.

We use the following notation throughout the text. We usually stick with the notation of having vectors denoted by lower-case bold letters; and, sets and subspaces being denoted with calligraphic fonts. For  $a < b$ ,  $[a : b]$  represents the set of numbers  $\{a, a + 1, \dots, b\}$ . (This is shortened as  $[b]$  for  $[1 : b]$ , and brackets are omitted in subscripts to improve readability.) The symbols stored at node  $i$  is represented by the vector  $\mathbf{s}_i$ , the symbols transmitted from node  $i$  to node  $j$  is denoted as  $\mathbf{d}_{i,j}$ , and the set  $\mathbf{d}_j$  is used to denote all of the downloaded symbols to node  $j$ . DSS is initialized with the  $n$  nodes containing encoded symbols, i.e.,  $\mathbf{s}_i = \mathbf{x}_i$  for  $i = 1, \dots, n$ .

### A. Cooperative repair in DSS

In most of the studies on DSS, exact repair for regenerating codes is analyzed in the context of single node failure. However, it is not uncommon to see simultaneous multiple node failures in storage networks, especially for large ones. The basic setup involves the simultaneous repair of  $t$  (possibly greater than one) failed nodes. After the failure of  $t$  storage nodes, the same number of newcomer nodes are introduced to the system. Each such node contacts to  $d$  live storage nodes and downloads  $\beta$  symbols from each of these nodes. In addition, utilizing a cooperative approach, each newcomer node also contacts other nodes being under repair and downloads  $\beta'$  symbols from each other node. Hence, the total repair cost is given by

$$\gamma = d\beta + (t - 1)\beta'. \quad (1)$$

Each newcomer node, to repair the  $i$ -th node of the original network, uses these  $d\beta + (t - 1)\beta'$  number of downloaded symbols to regenerate  $\alpha$  symbols,  $\mathbf{x}_i$ , and stores these symbols. This exact repair process preserves the MDS property, i.e., data stored on any  $k$  nodes (potentially including the nodes that are repaired) allows the original file  $\mathbf{f}$  to be reconstructed. See Fig. 1.

We remark that, as also argued in [21],  $d \geq k$  can be assumed without loss of generality. (Earlier papers on the subject assumed  $d \geq k$  case, and noted that this is assumed for simplicity. See, e.g., [16]–[20].) Remarkably, if  $d < k$ , a data collector can reconstruct the whole file by contacting only  $d$

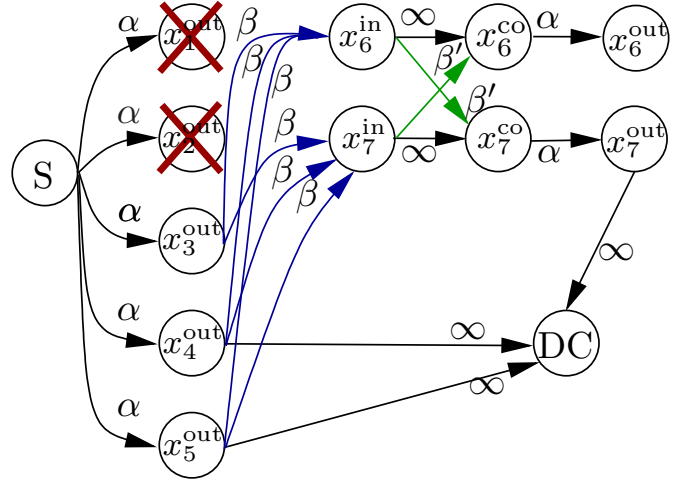


Fig. 1: Information flow graph of DSS implementing cooperative repair. In this representative example, we have  $n = 5$ ,  $d = k = 3$ , and  $t = 2$ . Accordingly, after a failure of two nodes, namely node 1 and node 2, the system cooperatively repairs these two nodes as node 6 and node 7. Downloads from live nodes (blue) and from cooperative repair pairs (green) are shown. Due to exact repair, the network will repair the nodes to satisfy  $x_6^{\text{out}} = x_1^{\text{out}}$  and  $x_7^{\text{out}} = x_2^{\text{out}}$ .

nodes, as from these nodes the other nodes can be repaired in groups of size  $t$ . Thus, any  $(n, k, d)$  code with  $d < k$  can be reduced to  $(n, k' = d, d)$  code. Therefore, without loss of generality, we will assume  $d \geq k$ .

### B. Information flow graph

In their seminal work [4], Dimakis et al. models the operation DSS using a multicasting problem over an information flow graph. (See Figs. 1 and 2 for the flow graph in the cooperative setting.) Information flow graph consists of three types of nodes:

- Source node ( $S$ ): Source node contains  $\mathcal{M}$  symbols long original file  $\mathbf{f}$ . The source node is connected to  $n$  nodes.
- Storage nodes  $((x_i^{\text{in}}, x_i^{\text{co}}, x_i^{\text{out}}))$ : In information flow graph associated with cooperative regenerating codes, we represent each node with a combination of three sub-nodes:  $x_i^{\text{in}}$ ,  $x_i^{\text{co}}$ , and  $x_i^{\text{out}}$ . Here,  $x_i^{\text{in}}$  is the sub-node having the connections from the live nodes,  $x_i^{\text{co}}$  is the sub-node having the connections from the nodes under repair in the same repair group, and  $x_i^{\text{out}}$  is the storage sub-node, which stores the data and is contacted by a data collector or other nodes under repair.  $x_i^{\text{in}}$  is connected to  $x_i^{\text{co}}$  with a link of infinite capacity,  $x_i^{\text{co}}$  is connected to  $x_i^{\text{out}}$  with a link of capacity  $\alpha$ . We represent cuts with a notation with bars as in  $(x_i^{\text{in}}, x_i^{\text{co}} | x_i^{\text{out}})$ , meaning the cut is passing through the link between  $x_i^{\text{co}}$  and  $x_i^{\text{out}}$ . (See Fig. 2.) The nodes on the right hand side of the cuts belong to data collector side, represented by the set  $\mathcal{D}$ , whereas the nodes belonging to the left hand side of the cuts belong to  $\mathcal{D}^c$ , the source side. For a newcomer node,  $x_i^{\text{in}}$  is connected to  $x_i^{\text{out}}$  sub-nodes of  $d$  live nodes with



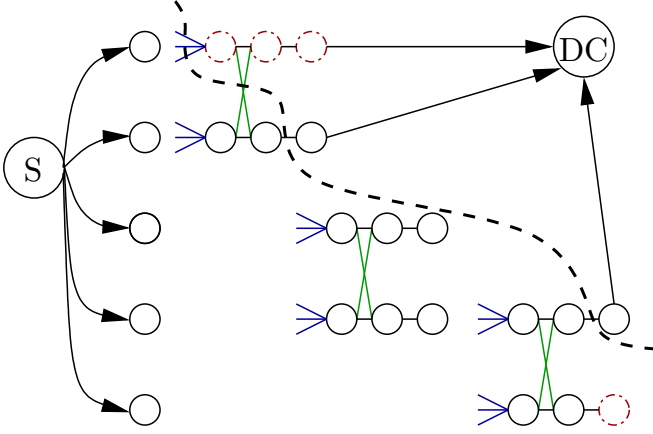


Fig. 2: Information flow graph of DSS implementing cooperative repair under security constraints. In this representative example, we have  $n = 5$ ,  $d = k = 3$ , and  $t = 2$ . Multiple repair stages and a cut, represented by dotted line, through the nodes connected to the DC are shown. The figure has different cut types: The first repaired node has a cut of type  $(|x^{\text{in}}, x^{\text{co}}, x^{\text{out}})$  and the second has a cut of type  $(x^{\text{in}}, x^{\text{co}}|x^{\text{out}})$ . Nodes that are being eavesdropped are indicated with dashed-dotted lines. Here, both the content and the downloads of the first repaired node is observed by the eavesdropper ( $\ell_2 = 1$ ), and only the content of the last repaired node is observed by the eavesdropper ( $\ell_1 = 1$ ). Accordingly, eavesdropper has observations of  $d\beta + (t-1)\beta'$  downloaded symbols from the first repaired node, and has  $\alpha$  number of symbols from the last repaired node.

links of capacity  $\beta$  symbols each, representing the data downloaded during node repair. This newcomer node also connects to  $x^{\text{in}}$  sub-nodes of  $(t-1)$  nodes being repaired in the same group, each having a link capacity of  $\beta'$ .

- Data collector node(s) (DC): Each data collector contacts  $x^{\text{out}}$  sub-node of  $k$  live nodes with edges each having  $\infty$ -link capacity.

### C. MBCR and MSCR points

With the aforementioned values of capacities of various edges in the information flow graph, the DSS is said to employ an  $(n, k, d, \alpha, \beta, \beta')$  code. For a given graph  $\mathcal{G}$  and data collectors  $\text{DC}_i$ , the file size that can be stored in such a DSS can be bounded using the max flow-min cut theorem for multicasting utilized in network coding [42], [43].

**Lemma 1** (Max flow-min cut theorem for multicasting [4], [42], [43]).

$$\mathcal{M} \leq \min_{\mathcal{G}} \min_{\text{DC}_i} \text{maxflow}(S \rightarrow \text{DC}_i, \mathcal{G}),$$

where  $\text{flow}(S \rightarrow \text{DC}_i, \mathcal{G})$  represents the flow from the source node  $S$  to data collector  $\text{DC}_i$  over the graph  $\mathcal{G}$ .

Therefore, e.g., for the graph in Fig. 2,  $\mathcal{M}$  symbol long file can be delivered to a data collector DC, only if the min cut is at least  $\mathcal{M}$ .

Dimakis et al., [4], consider  $k$  successive node failures and evaluate the min-cut over possible graphs, and obtain the following bound (for  $t = 1$  case).

$$\mathcal{M} \leq \sum_{i=0}^{k-1} \min\{\alpha, (d-i)\beta\} \quad (2)$$

We emphasize that the min-cut for this ( $t = 1$ ) case is given by the scenario where  $k$  successively repaired nodes are connected to DC, and, for each successive repair, the repaired node  $i+1$  also connects to  $i$  number of previously repaired nodes. Hence, for each DC-connected node, the cut value is equal to  $(d-i)\beta$  if the cut is of type  $(|x^{\text{in}}, x^{\text{out}})$ , and is equal to  $\alpha$  if the cut is of type  $(x^{\text{in}}|x^{\text{out}})$ . (Note that,  $x^{\text{co}}$  does not appear here as the model considered in [4] does not involve cooperative repair.) The codes that attain the bound in (2) are named as regenerating codes [4].

For the cooperative scenario, we consider secure file size upper bound in the next section using similar min cut arguments in the presence of eavesdroppers. Removing the leakage (to eavesdropper) terms one will obtain the min cut file size bound for the cooperative scenario. In particular, a file size bound in the cooperative setting is obtained as follows.

$$\mathcal{M} \leq \sum_{i=0}^{g-1} u_i \min \left\{ \alpha, \left( d - \sum_{j=0}^{i-1} u_j \right) \beta + (t - u_i) \beta' \right\}, \quad (3)$$

where  $u_i \in [0 : t]$  is the number of repaired nodes in repair group  $i \in [0 : g-1]$  that is connected to DC. Similar to the  $t = 1$  case described above, the cut of type  $(x^{\text{in}}, x^{\text{co}}|x^{\text{out}})$  has a value of  $\alpha$ . The cut of type  $(|x^{\text{in}}, x^{\text{co}}, x^{\text{out}})$ , on the other hand, has a value of  $(t - u_i)\beta'$  due to the links coming from the nodes under repair that are not connected to DC and additional value of  $(d - \sum_{j=0}^{i-1} u_j)\beta$  due to the connections to the previously repaired live nodes that are not contacted by DC. (Here, we again subtract the values of the flows from the nodes already belonging to the data collector side,  $\mathcal{D}$ .) The cut of type  $(x^{\text{in}}|x^{\text{co}}, x^{\text{out}})$  has value of  $\infty$  and hence, does not appear in the min-cut.

Note that, given a file size  $\mathcal{M}$ , there is an inherent trade off between storage per node  $\alpha$  and repair bandwidth  $\gamma \triangleq d\beta + (t-1)\beta'$ . This trade off, for the cooperative setting, can be established using a similar analyses leading to MBR/MSR points from the equation (2). Two classes of codes that achieve two extreme points of this trade off are named as *minimum bandwidth cooperative regenerating (MBCR)* codes and *minimum storage cooperative regenerating (MSCR)* codes. The former is obtained by first finding the minimum possible  $\gamma$  and then finding the minimum  $\alpha$  satisfying (3). This point is given by the following.

$$\begin{aligned} \alpha_{\text{MBCR}} &= \frac{\mathcal{M}}{k} \frac{2d+t-1}{2d+t-k}, & \gamma_{\text{MBCR}} &= \alpha_{\text{MBCR}}, \\ \beta_{\text{MBCR}} &= \frac{\mathcal{M}}{k} \frac{2}{2d+t-k}, & \beta'_{\text{MBCR}} &= \frac{\mathcal{M}}{k} \frac{1}{2d+t-k} \end{aligned} \quad (4)$$

MSCR point, on the other hand, is obtained by first choosing a minimum storage per node (i.e.,  $\alpha = \mathcal{M}/k$ ), and then

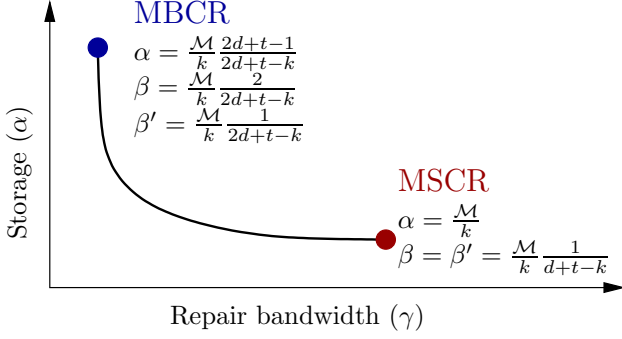


Fig. 3: Storage vs. repair bandwidth trade off for cooperative regenerating codes. The repair bandwidth is given by  $\gamma = d\beta + (t-1)\beta'$ .

minimizing  $\gamma$  (via choosing minimum possible  $\beta$ - $\beta'$  pair) satisfying the min cut (3).

$$\begin{aligned} \alpha_{\text{MSCR}} &= \frac{\mathcal{M}}{k}, & \gamma_{\text{MSCR}} &= \frac{\mathcal{M}}{k} \frac{d+t-1}{d+t-k}, \\ \beta_{\text{MSCR}} &= \frac{\mathcal{M}}{k} \frac{1}{d+t-k}, & \beta'_{\text{MSCR}} &= \frac{\mathcal{M}}{k} \frac{1}{d+t-k} \end{aligned} \quad (5)$$

We depict these two trade off points, which are directly computable from (3), in Fig. 3. (We refer reader to the works [13], [14] for a detailed derivation of these two points. See also [15] for an analysis for the simplified case of when  $t|k$ , i.e., the number of groups satisfies  $g = k/t$ .) Note that, when  $t = 1$ , these two points correspond to MBR/MSR points characterized in [4].

#### D. Eavesdropper model

We consider an  $(\ell_1, \ell_2)$  eavesdropper, which can access the stored data of nodes in the set  $\mathcal{E}_1$ , and additionally can access both the stored and downloaded data at the nodes in the set  $\mathcal{E}_2$ , where  $\ell_1 = |\mathcal{E}_1|$  and  $\ell_2 = |\mathcal{E}_2|$ . Hence, the eavesdropper has access to  $x_i^{\text{out}}$  for  $i \in \mathcal{E}_1$  and  $x_j^{\text{in}}, x_j^{\text{co}}, x_j^{\text{out}}$  for  $j \in \mathcal{E}_2$ . (See Fig. 2.) This is the eavesdropper model defined in [27] (adapted here to the cooperative repair setting), which generalizes the eavesdropper model considered in [5]. The eavesdropper is assumed to know the coding scheme employed by the DSS. At the MBCR point, a newcomer downloads  $\alpha_{\text{MBCR}} = \gamma_{\text{MBCR}}$  amount of data. Thus, an eavesdropper does not gain any additional information if it is allowed to access the data downloaded during repair. However, at the MSCR point, repair bandwidth is strictly greater than the per node storage,  $\alpha_{\text{MSCR}}$ , and an eavesdropper potentially gains more information if it has access to the data downloaded during node repair as well. We summarize the eavesdropper model together with the definition of achievability of a secure file size in the following.

**Definition 2** (Security against an  $(\ell_1, \ell_2)$  eavesdropper). A DSS is said to achieve a secure file size of  $\mathcal{M}^s$  against an  $(\ell_1, \ell_2)$  eavesdropper, if, for any sets  $\mathcal{E}_1$  and  $\mathcal{E}_2$  of size  $\ell_1$  and  $\ell_2$ , respectively,  $I(\mathbf{f}^s; \mathbf{e}) = 0$ . Here  $\mathbf{f}^s$  is the secure file of size  $\mathcal{M}^s$ , which is first encoded to file  $\mathbf{f}$  of size  $\mathcal{M}$  before storing

into DSS, and  $\mathbf{e}$  is the eavesdropper observation vector given by  $\mathbf{e} \triangleq \{x_i^{\text{out}}, x_j^{\text{in}}, x_j^{\text{co}}, x_j^{\text{out}} : i \in \mathcal{E}_1, j \in \mathcal{E}_2\}$ .

We remark that, as it will be clear from the following sections, when a file  $\mathbf{f}$  of size  $\mathcal{M}$  is stored in DSS and the secure file size achieved is  $\mathcal{M}^s$ , the remaining  $\mathcal{M} - \mathcal{M}^s$  symbols can be utilized as public data, which does not have security constraints. Yet, noting the possibility of storing the public data, we will refer to this uniformly distributed part as the random data, which is utilized to achieve security. Finally, we note the following lemma, which will be used in the following parts of the sequel.

**Lemma 3** (Secrecy Lemma). Consider a system with information bits  $\mathbf{u}$ , random bits  $\mathbf{r}$  (independent of  $\mathbf{u}$ ), and an eavesdropper with observations given by  $\mathbf{e}$ . If  $H(\mathbf{e}) \leq H(\mathbf{r})$  and  $H(\mathbf{r}|\mathbf{u}, \mathbf{e}) = 0$ , then  $I(\mathbf{u}; \mathbf{e}) = 0$ .

*Proof:* See Appendix A. ■

### III. SECURE MBCR CODES

In this section, we study secure minimum bandwidth cooperative regenerating codes. We first present an upper bound on the secure file size that can be supported by an MBCR code. Then, we present exact repair coding schemes achieving the derived bound. In addition, we analyze how the cooperation affects the penalty paid in securing storage systems.

#### A. Upper bound on secure file size of MBCR codes

Analysis of the cut-set bounds for cooperative regenerating codes are provided in [13], [14]. (See also the arguments given in [11], [15]. Here, we follow the notations of [13], [15].) We consider groups of nodes being repaired, and denote the number of nodes in group  $i$  that are repaired in group  $i$  and contacted by the data collector as  $u_i$  such that

$$\begin{aligned} u_i &\in [t], \forall i = 0, 1, \dots, g-1, \\ \sum_{i=0}^{g-1} u_i &= k, \end{aligned}$$

where  $g$  is the total number of groups that have been repaired. While evaluating an upper bound on the file size that can be securely stored on the DSS, the data collector under consideration is assumed to contact only these  $k$  nodes that belong to one of these  $g$  groups.

We consider two types of cuts:  $m_i$  number of nodes have the first cut type  $(x^{\text{in}}, x^{\text{co}}|x^{\text{out}})$ , and  $u_i - m_i$  number of nodes have the second cut type  $(x^{\text{in}}, x^{\text{co}}, x^{\text{out}})$ ,  $0 \leq i \leq g-1$ . Note that the cuts of the form  $(x^{\text{in}}, x^{\text{co}}|x^{\text{out}})$  give a cut value of  $\alpha$  as opposed to  $(x^{\text{in}}|x^{\text{co}}, x^{\text{out}})$ , which has cut value larger than  $\alpha$ . Since we are interested in the cuts of smaller size, we do not consider the cuts  $(x^{\text{in}}|x^{\text{co}}, x^{\text{out}})$ .

We consider  $\ell_1$  number of colluding eavesdroppers, each observing the contents of different nodes. Note that, for MBCR point analysis, we can consider  $\ell_2 = 0$  without loss of generality, as the amount of data a particular node stores is equal to amount of data it downloads during its repair. We denote the number of eavesdroppers on the nodes in the first cut type as  $\ell_1^{i,1}$ ,  $0 \leq i \leq g-1$ ; and denote the number

of eavesdroppers on the nodes in the second cut type as  $l_1^{i,2}, 0 \leq i \leq g-1$ , such that

$$\begin{aligned} l_1^{i,1} &\leq m_i \\ l_1^{i,2} &\leq u_i - m_i \\ \sum_{i=0}^{g-1} (l_1^{i,1} + l_1^{i,2}) &= \ell_1. \end{aligned}$$

Thus, for group  $i$ , due to the eavesdroppers, the nodes that belong to the first type can only add the value of  $(m_i - l_1^{i,1})\alpha$  to the cut. The second type, on the other hand, consists of  $u_i - m_i$  nodes, out of which  $l_1^{i,2}$  of them are eavesdropper. As the data downloaded is equal to the data stored at MBCR point, the nodes that are eavesdropped do not add a value to the cut. The remaining  $u_i - m_i - l_1^{i,2}$  number of nodes contact  $d$  live nodes,  $\sum_{j=0}^{i-1} u_j$  number of these belong to the previous groups being repaired. In addition, these nodes contact  $t-1$  nodes from the same repair group, out of which  $u_i - m_i - 1$  number of nodes belong to  $\mathcal{D}$ . Accordingly, this cut-set bound is given by the following.

$$\mathcal{M}^s \leq \sum_{i=0}^{g-1} \left( (m_i - l_1^{i,1})\alpha + (u_i - m_i - l_1^{i,2})C_i \right), \quad (6)$$

where

$$C_i = \left( d - \sum_{j=0}^{i-1} u_j \right) \beta + (t - u_i + m_i) \beta'.$$

Each summation term in (6) is concave for  $m_i \in [0, u_i]$ . We consider two scenarios in (6), (i)  $m_i = 0, l_1^{i,2} = l_1^i$  and (ii)  $m_i = u_i, l_1^{i,1} = l_1^i$ . Hence, we obtain,

$$\mathcal{M}^s \leq \sum_{i=0}^{g-1} (u_i - l_1^i) \min \left\{ \alpha, \left( d - \sum_{j=0}^{i-1} u_j \right) \beta + (t - u_i) \beta' \right\}. \quad (7)$$

Note that, at MBCR point, the nodes store what they download, therefore the MBCR codes should satisfy

$$\alpha = d\beta + (t-1)\beta'. \quad (8)$$

Utilizing this, we consider the following cases of (7).

**Case 1:**  $g = k, u_i = 1, \forall i = 0, \dots, k-1$

$$\mathcal{M}^s \leq \sum_{i=0}^{k-1} (1 - l_1^i) ((d-i)\beta + (t-1)\beta') \quad (9)$$

Here, the minimum cut value corresponds to having  $l_1^i = 1$  for  $i = 0, 1, \dots, \ell_1 - 1$ ; and  $l_1^i = 0$  otherwise. Hence, we get

$$\mathcal{M}^s \leq \sum_{i=\ell_1}^{k-1} (d-i)\beta + (t-1)\beta', \quad (10)$$

from which we obtain

$$\mathcal{M}^s \leq \frac{(k - \ell_1)(2d - k - \ell_1 + 1)}{2} \beta + (k - \ell_1)(t-1)\beta'. \quad (11)$$

**Case 2:** If  $t \geq k, g = 1, u_0 = k$

$$\mathcal{M}^s \leq (k - \ell_1)(d\beta + (t-k)\beta') \quad (12)$$

**Case 3:** If  $t < k, g = \lfloor k/t \rfloor + 1, u_i = t$  for  $i = 0, \dots, g-2$ , and  $u_{g-1} = k - \lfloor k/t \rfloor t$

Let  $a \triangleq \lfloor k/t \rfloor$  and  $b \triangleq k - at$ , so that  $k = at + b$ . From (7), we obtain

$$\mathcal{M}^s \leq \sum_{i=0}^{a-1} (t - l_1^i)(d - it)\beta + (b - l_1^a) \{ (d - at)\beta + (t - b)\beta' \}. \quad (13)$$

Considering possible allocations of eavesdroppers in this bound, i.e.,  $\{l_1^i\}_{i=0}^{g-1}$ , we obtain the following bound (where we collect eavesdropper dependent terms in the variable  $S$  given below).

$$\mathcal{M}^s \leq \beta \left\{ kd + \frac{(k-b)(t-k-b)}{2} \right\} + \beta' b(t-b) - S, \quad (14)$$

where  $S$  is given by

$$S \triangleq \max_{l_1^i \leq t \text{ s.t. } \sum_{i=0}^a l_1^i = \ell_1} \sum_{i=0}^{a-1} l_1^i (d - it)\beta + l_1^a \{ (d - at)\beta + (t - b)\beta' \} \quad (15)$$

$$= \begin{cases} \sum_{i=0}^{\lfloor \ell_1/t \rfloor - 1} t(d - it)\beta + (\ell_1 - \lfloor \ell_1/t \rfloor t)(d - \lfloor \ell_1/t \rfloor t)\beta \\ = \beta \ell_1 (d - \lfloor \ell_1/t \rfloor t) + \frac{t^2 \beta}{2} \lfloor \ell_1/t \rfloor (\lfloor \ell_1/t \rfloor + 1), \\ \text{if } \ell_1 \leq at = k - b. \\ \sum_{i=0}^{a-1} t(d - it)\beta + (\ell_1 - at) \{ (d - at)\beta + (\ell_1 - at)(t - b)\beta' \} \\ = \beta \ell_1 (d - at) + \frac{t^2 \beta}{2} a(a+1) + (\ell_1 - at)(t - b)\beta', \\ \text{if } \ell_1 \geq at = k - b. \end{cases}$$

Note that we consider the worst case eavesdropper allocation to maximize  $S$  in the above derivation.

The normalized values at the MBCR point are given by

$$\beta' = 1, \beta = 2, \alpha = \gamma = 2d + t - 1, \mathcal{M} = k(2d - k + t). \quad (16)$$

Using this and the bounds given in (11), (12), and (14), we get a bound on the secure file size at the MBCR point. We state this result in the following.

**Proposition 4.** Cooperative regenerating codes operating at the MBCR point with a secure file size of  $\mathcal{M}^s$  satisfy

$$\begin{aligned} \mathcal{M}^s &\leq k(2d - k + t) - \ell_1(2d - \ell_1 + t) \\ &= (k - \ell_1)(2d + t - k - \ell_1), \end{aligned} \quad (17)$$

and the MBCR point is given by  $\beta' = 1$ ,  $\beta = 2$ ,  $\alpha = \gamma = 2d + t - 1$  for a file size of  $\mathcal{M} = k(2d - k + t)$ .

*Proof:* We show that (12) and (14) result in loose bounds compared to that of (11) in Appendix B. And, (11) evaluates to the stated bound at the MBCR point. ■

### B. Code construction for secure MBCR when $n = d + t$

We consider secrecy precoding of the data at hand before storing it to DSS nodes using an MBCR code. We establish this precoding with maximum rank distance (MRD) codes. In vector representation, assuming  $m \geq n$ , the norm of a vector  $\mathbf{v} \in \mathbb{F}_{q^m}^n$  is the column rank of  $\mathbf{v}$  over the base field  $\mathbb{F}_q$ , denoted by  $Rk(\mathbf{v}|\mathbb{F}_q)$ . (This is the maximum number of linearly independent coordinates of  $\mathbf{v}$  over the base field  $\mathbb{F}_q$ , for a given basis of  $\mathbb{F}_{q^m}$  over  $\mathbb{F}_q$ . A basis also establishes an isomorphism between  $n$ -length vectors, in  $\mathbb{F}_{q^m}^n$ , to  $m \times n$  matrices, in  $\mathbb{F}_q^{m \times n}$ .) Rank distance between two vectors is defined by  $d(\mathbf{v}_1, \mathbf{v}_2) = Rk(\mathbf{v}_1 - \mathbf{v}_2|\mathbb{F}_q)$ . (In matrix representation, this is equivalent to the rank of the difference of the two corresponding matrices of the vectors.) An  $[n, k, d]$  MRD code over the extension field  $\mathbb{F}_{q^m}$  achieving the maximum rank distance  $d = n - k + 1$  (for  $m \geq n$ ) can be constructed with the following linearized polynomial. (This is referred to as the Gabidulin construction of MRD codes, or Gabidulin codes [44]–[47].)

$$f(g) = \sum_{i=0}^{k-1} u_i g^{[i]}, \quad (18)$$

where  $[i] = q^i$ , and  $g, u_i \in \mathbb{F}_{q^m}$ . Then, given  $n$  linearly independent elements over  $\mathbb{F}_q$ ,  $\{g_1, \dots, g_n\}$  with  $g_j \in \mathbb{F}_{q^m}$ , the codewords for a given set of  $k$  elements,  $u_i \in \mathbb{F}_{q^m}$ ,  $i = [0 : k-1]$ , are obtained by  $x_j = f(g_j) = \sum_{i=0}^{k-1} u_i g_j^{[i]}$  for  $j = [1 : n]$ . (With generator matrix representation, we have  $\mathbf{x} = \mathbf{u}\mathbf{G}$ , where  $\mathbf{G} = [g_1, \dots, g_n; \dots; g_1^{[k-1]}, \dots, g_n^{[k-1]}]$ .) We also note that the linearized polynomial satisfies  $f(a_1 g_1 + a_2 g_2) = a_1 f(g_1) + a_2 f(g_2)$ , for a given  $a_1, a_2 \in \mathbb{F}_q$  and  $g_1, g_2 \in \mathbb{F}_{q^m}$ , and this will be utilized in the following.

Consider now the MBCR point given by  $\mathcal{M} = k(2d - k + t)$ ,  $\beta' = 1$ ,  $\beta = 2$ ,  $\alpha = \gamma = 2d + t - 1$ ,  $\mathcal{M}^s = k(2d - k + t) - \ell_1(2d - \ell_1 + t)$ , and  $n = d + t$ . We use MRD codes with  $n = k = \mathcal{M}$ ; hence, the rank distance bound  $d \leq n - k + 1$  is saturated at  $d = 1$ . Accordingly, we utilize  $[\mathcal{M}, \mathcal{M}, 1]$  MRD codes over  $\mathbb{F}_{q^m}$ , which maps length  $\mathcal{M}$  vectors (each element of it being in  $\mathbb{F}_{q^m}$ ) to length  $\mathcal{M}$  codewords in  $\mathbb{F}_{q^m}^{\mathcal{M}}$  (with  $m \geq \mathcal{M}$ ). The coefficients of the underlying linearized polynomial ( $f(g)$ ) are chosen by  $\mathcal{M} - \mathcal{M}^s$  random symbols denoted by  $\mathbf{r} \in \mathbb{F}_{q^m}^{\mathcal{M} - \mathcal{M}^s}$  and  $\mathcal{M}^s$  secure data symbols denoted by  $\mathbf{u} \in \mathbb{F}_{q^m}^{\mathcal{M}^s}$ . The corresponding polynomial  $f(g)$  is evaluated at  $\mathcal{M}$  points  $\{g_1, \dots, g_{\mathcal{M}}\}$ , which are linearly independent over  $\mathbb{F}_q$ . We denote these as  $x_j = f(g_j)$  for  $j = 1, \dots, \mathcal{M}$ . This finalizes the secrecy precoding step.

The second encoding step is based on the encoding scheme for cooperative repair proposed in [20]. (Here, we will summarize file recovery and node repair processes for the case of MRD precoding, and provide the proof of

security.) Split the  $\mathcal{M}$  symbols into two parts a)  $x_1$  to  $x_{nk}$ , and b)  $x_{nk+1}$  to  $x_{nk+k(d-k)}$ . (Note that  $n = d + t$  and  $\mathcal{M} = nk + k(d - k)$ .) The first part is divided into  $n$  groups of  $k$  symbols, and stored in  $n$  nodes. Here, node  $i$  stores  $x_{(i-1)k+1}$  to  $x_{ik}$ . The second part is divided into  $d - k$  groups of  $k$  symbols. These symbols are encoded with an  $(n, k)$  MDS code, and stored on  $n$  nodes. In particular,  $\{y_{j,1}, \dots, y_{j,n}\}$  are generated from symbols  $\{x_{nk+(j-1)k+1}, \dots, x_{nk+jk}\}$ , and  $y_{j,i}$  is stored at node  $i$ , for  $j = 1, \dots, d - k$ . Node  $i$ , having stored  $\{x_{(i-1)k+1}, \dots, x_{ik}, y_{1,i}, \dots, y_{d-k,i}\}$ , which is referred to as the primary data of node  $i$ , encodes these symbols using an  $(n-1, d)$  MDS code having a Vandermonde matrix  $\Phi$  of size  $d \times (n-1)$  as its generator matrix. (This choice of  $\Phi$  ensure that  $[\mathbf{I}_d \Phi]$  is generator matrix for an  $(n + d - 1, d)$  MDS code.) These  $n - 1$  symbols are stored in every other node one-by-one. We denote the encoded primary data of node  $i$  that is stored in node  $j \neq i$  as  $z_{j,i}$ . We call these as the secondary data. This procedure is repeated for every node, so that each node  $i$  stores  $\{x_{(i-1)k+1}, \dots, x_{ik}, y_{1,i}, \dots, y_{d-k,i}, z_{i,1}, \dots, z_{i,i-1}, z_{i,i+1}, \dots, z_{i,n}\}$ , and hence total number of symbols stored at each node is  $k + (d - k) + (n - 1) = d + n - 1 = 2d + t - 1 = \alpha$ .

*File recovery at DC:* DC connects to any  $k$  nodes, without loss of generality we assume the first  $k$  nodes. From  $y_{j,1:k}$ , DC can obtain  $x_{nk+(j-1)k+1}, \dots, x_{nk+jk}$ , for each  $j = [1 : d - k]$ . It can re-encode this into  $y_{j,1:n}$  using the MDS code, and obtain the other  $y$  symbols at the remaining nodes. Then, for each  $i \in [k + 1 : n]$ , DC can use the MDS property of  $[\mathbf{I}_d \Phi]$ , to obtain  $x_{(i-1)k+1}, \dots, x_{ik}$  symbols of node  $i$  from the  $k$  secondary data symbols of the contacted nodes, i.e.,  $z_{j,i}$  for  $j = [1 : k]$ , and additional  $d - k$  symbols,  $y_{j,i}$  for  $j = [1 : d - k]$ . Having obtained  $x_1, \dots, x_{\mathcal{M}}$ , DC can perform interpolation to solve for both data and random coefficients.

*Node repair:* Assume that the first  $t$  nodes fail. From the secondary data stored in the remaining  $d = n - t$  nodes,  $z_{t+1,i}, \dots, z_{n,i}$ , one can recover  $x_{(i-1)k+1}, \dots, x_{ik}$  and  $y_{1,i}, \dots, y_{d-k,i}$  for node  $i = 1, \dots, t$ . (This corresponds to sending 1 symbol from each of  $d$  nodes to each of the  $t$  nodes.) Then, to recover the secondary data stored at each node under repair, say for the node  $j = 1, \dots, t$ , every other node, i.e., nodes  $i \neq j$ , including the nodes under repair, computes and sends its corresponding encoded primary data, i.e.,  $z_{j,i}$ , to node  $j$ . (This corresponds to sending 1 symbol from each node to each of the  $t$  nodes.) This achieves  $\beta = 2$  and  $\beta' = 1$  symbols for the repair procedure.

*Security:* Consider that the eavesdropper is observing the first  $\ell_1$  nodes. Due to the code construction, the symbols in the sets  $\mathcal{X} = \{x_1, \dots, x_{\ell_1 k}\}$ ,  $\mathcal{Y} = \{y_{1,1}, \dots, y_{d-k,1}, \dots, y_{1,\ell_1}, \dots, y_{d-k,\ell_1}\}$ ,  $\mathcal{Z} = \{z_{j,i} \text{ for } j = 1, \dots, \ell_1, \text{ and } i = \ell_1 + 1, \dots, n\}$  correspond to linearly independent evaluation points. (Note that, the symbols  $\{z_{j,i}\}$  for  $j = 1, \dots, \ell_1$ ;  $i = 1, \dots, \ell_1$ ;  $j \neq i$ , are linear combinations of the symbols in  $\mathcal{X} \cup \mathcal{Y}$ .) Due to the linearized property of the code, the eavesdropper observing  $\ell_1 \alpha = \ell_1(2d + t - 1)$  symbols, has evaluation of polynomial  $f(\cdot)$  at  $\ell_1(2d + t - \ell_1)$  linearly independent points. Using the data symbols, together with interpolation from these  $\ell_1(2d + t - \ell_1)$  symbols, the eavesdropper can solve for  $\ell_1(2d + t - \ell_1)$  random symbols.



Therefore, denoting the eavesdroppers' observation as  $\mathbf{e}$ , we have  $H(\mathbf{r}|\mathbf{e}, \mathbf{u}) = 0$ . As,  $H(\mathbf{e}) = H(\mathbf{r})$ , from Lemma 3, we have  $I(\mathbf{u}; \mathbf{e}) = 0$ .

Using the upper bound given in Proposition 4, we obtain the following result.

**Proposition 5.** *The secrecy capacity at MBCR point for a file size of  $\mathcal{M}^s = k(2d - k + t)$  is given by  $\mathcal{M}^s = k(2d - k + t) - \ell_1(2d - \ell_1 + t)$ , if  $n = d + t$ .*

### C. Does cooperation enhances/degrades security at MBCR?

Cooperative regenerating codes has a repair bandwidth given by  $\gamma = d\beta + (t - 1)\beta'$ . In this section, we analyze  $\frac{\gamma}{\mathcal{M}^s}$ , the ratio of repair bandwidth to the secure file size. In the following, we refer to this parameter as the normalized repair bandwidth (NRBW).

Without the security constraints, for which  $\ell_1 = 0$  in Proposition 4, we observe that at MBCR point NRBW is given by

$$\text{NRBW}(\ell_1 = 0) = \frac{2d + t - 1}{k(2d - k + t)}, \quad (19)$$

which is equal to

$$\text{NRBW}(\ell_1 = 0, n = d + t) = \frac{2n - t - 1}{k(2n - k - t)} \quad (20)$$

for a system with  $n = d + t$ . Here, the classical (i.e., non-cooperative) scenario corresponds to  $t = 1$  case, which has an NRBW of

$$\text{NRBW}(\ell_1 = 0, n = d + t, t = 1) = \frac{2n - 2}{k(2n - k - 1)}. \quad (21)$$

Comparing the last two equations, we see that

$$\text{NRBW}(\ell_1 = 0, n = d + t) \geq \text{NRBW}(\ell_1 = 0, n = d + t, t = 1),$$

with equality iff  $t = 1$ . Therefore, without the security constraints, having simultaneous repairs of size greater than 1 actually increases the repair bandwidth. This nature of cooperation also results in the conclusion that deliberately delaying the repairs does not bring additional savings [13]. (This observation is proposed for both MBCR and MSCR points in [13] with an analysis of derivative of  $\gamma$  with respect to  $t$ . Here, we provide an analysis with NRBW.)

We revisit the above conclusion under security constraints. The question is whether the cooperation (i.e., having a system with multiple failures, or deliberately delaying the repairs) results in a loss/gain in secure DSS. A calculation similar to above shows that NRBW for the case  $t > 1$  is strictly greater than that of  $t = 1$  when  $n = d + t$  for  $\ell_1 < k$ . The MBCR points given in Proposition 4 for codes satisfying  $0 \leq \ell_1 < k < n$ ,  $d \geq k$ , and  $d = n - t$  are given in Table I in Appendix C. As evident from the table, we see that cooperation does not bring additional savings for secure DSS at MBCR point when  $d + t = n$ . This in turn means that one may not delay the repairs to achieve a better performance than that of single failure-repair if  $d$  is chosen such that  $n = d + t$  for a given  $t, n$ . However, if the downloads within the cooperative group are less costly compared to the downloads from the live nodes, then delaying repairs would be beneficial in reducing the total cost. We will revisit this analysis for codes having  $n > d + t$  in the next subsection.

### D. General Code Construction for Secure MBCR

The code construction above needs the requirement of  $d = n - t$ . However, for practical systems, it may not be possible that a failed node connects to all the remaining nodes. This brings the necessity of code constructions for  $d < n - t$ . Remarkably, for a fixed  $(n, k, d, \mathcal{M})$ , increasing  $t$  can reduce the repair bandwidth in the secrecy scenario we consider here. This is reported in [16] for DSS without secrecy constraints. Hence, for a fixed  $d$ , delaying the repairs can be advantageous, e.g., when there is a limit on the number of live nodes that can be connected. In the following, we present a general construction which works for any parameters, in particular for  $n > d + t$ .

The construction is based on the code construction proposed in [21]. In [21], a bivariate polynomial is constructed using  $\mathcal{M} = k(2d + t - k)$  message symbols as the coefficients of the polynomial:

$$\begin{aligned} F(X, Y) = & \sum_{\substack{0 \leq i < k, \\ 0 \leq j < k}} a_{ij} X^i Y^j + \sum_{\substack{0 \leq i < k, \\ k \leq j < d+t}} b_{ij} X^i Y^j \\ & + \sum_{\substack{k \leq i < d, \\ 0 \leq j < k}} c_{ij} X^i Y^j \end{aligned} \quad (22)$$

Given  $q > n$ , two set of  $n$  distinct points,  $\{x_1, x_2, \dots, x_n\}$  and  $\{y_1, y_2, \dots, y_n\}$ , are chosen. The  $i^{\text{th}}$  node in the DSS store the following  $2d + t - 1$  evaluations of polynomial  $F(X, Y)$ :

$$\begin{aligned} & F(x_i, y_i), F(x_i, y_{i \oplus 1}), \dots, F(x_i, y_{i \oplus (d+t-1)}) \\ & F(x_{i \oplus 1}, y_i), F(x_{i \oplus 2}, y_i), \dots, F(x_{i \oplus (d-1)}, y_i) \end{aligned} \quad (23)$$

where  $\oplus$  denotes addition modulo  $n$ . The first  $d + t$  evaluation at node  $i$  can be seen as the evaluation of univariate polynomial  $f_i(Y) = F(x_i, Y)$  of degree at most  $d + t - 1$  at  $d + t$  points. This uniquely defines the polynomial  $f_i(Y)$ . Similarly, the first evaluation in (23),  $F(x_i, y_i)$ , along with last  $d - 1$  evaluations uniquely define the univariate polynomial  $g_i(X) = F(X, y_i)$  of degree at most  $d - 1$ . This property of the proposed bivariate polynomial based coding scheme is utilized for the exact node repair and data reconstruction processes at MBCR point. (We refer to [21] for details.)

In order to get an  $(\ell_1, 0)$  secure code at MBCR point, we rewrite the polynomial in (22) as follows:

$$\begin{aligned} F(X, Y) = & \sum_{\substack{0 \leq i < \ell_1, \\ 0 \leq j < \ell_1}} a_{ij} X^i Y^j + \sum_{\substack{0 \leq i < \ell_1, \\ \ell_1 \leq j < k}} a_{ij} X^i Y^j \\ & + \sum_{\substack{\ell_1 \leq i < k, \\ 0 \leq j < \ell_1}} a_{ij} X^i Y^j + \sum_{\substack{\ell_1 \leq i < k, \\ \ell_1 \leq j < k}} a_{ij} X^i Y^j \\ & + \sum_{\substack{0 \leq i < \ell_1, \\ k \leq j < d+t}} b_{ij} X^i Y^j + \sum_{\substack{\ell_1 \leq i < k, \\ k \leq j < d+t}} b_{ij} X^i Y^j \\ & + \sum_{\substack{k \leq i < d, \\ 0 \leq j < \ell_1}} c_{ij} X^i Y^j + \sum_{\substack{k \leq i < d, \\ \ell_1 \leq j < k}} c_{ij} X^i Y^j \end{aligned} \quad (24)$$

Next, we choose  $\ell_1^2 + \ell_1(k - \ell_1) + (k - \ell_1)\ell_1 + \ell_1(d + t - k) + (d - k)\ell_1 = \ell_1(2d + t - \ell_1)$  coefficients of  $F(X, Y)$ ,



$F(x_1, y_1)$	$F(x_1, y_2)$	$\dots$	$F(x_1, y_{\ell_1})$	$\dots$	$F(x_1, y_{d+t})$			
$F(x_2, y_1)$	$F(x_2, y_2)$	$\dots$	$F(x_2, y_{\ell_1})$	$\dots$	$F(x_2, y_{d+t})$	$\mathbf{F}(\mathbf{x}_2, \mathbf{y}_{d+t+1})$		
$\dots$								
$F(x_{\ell_1}, y_1)$	$F(x_{\ell_1}, y_2)$	$\dots$	$F(x_{\ell_1}, y_{\ell_1})$	$\dots$	$F(x_{\ell_1}, y_{d+t})$	$\mathbf{F}(\mathbf{x}_{\ell_1}, \mathbf{y}_{d+t+1})$	$\dots$	$\mathbf{F}(\mathbf{x}_{\ell_1}, \mathbf{y}_{d+t-1+\ell_1})$
$\dots$								
$F(x_d, y_1)$	$F(x_d, y_2)$	$\dots$	$F(x_d, y_{\ell_1})$					
	$\mathbf{F}(\mathbf{x}_{d+1}, \mathbf{y}_2)$	$\dots$	$\mathbf{F}(\mathbf{x}_{d+1}, \mathbf{y}_{\ell_1})$					
		$\dots$						
			$\mathbf{F}(\mathbf{x}_{d+\ell_1-1}, \mathbf{y}_{\ell_1})$					

Fig. 4: Observed symbols at the eavesdroppers for a given  $\ell_1$ .

$\{a_{ij}\}_{0 \leq i < \ell_1, 0 \leq j < \ell_1}$ ,  $\{a_{ij}\}_{0 \leq i < \ell_1, \ell_1 \leq j < k}$ ,  $\{a_{ij}\}_{\ell_1 \leq i < k, 0 \leq j < \ell_1}$ ,  $\{b_{ij}\}_{0 \leq i < \ell_1, k \leq j < d+t}$ ,  $\{c_{ij}\}_{k \leq i < d, 0 \leq j < \ell_1}$ , to be random symbols drawn from  $\mathbb{F}_q$  in an i.i.d. manner. Remaining  $k(2d+t-k) - \ell_1(2d+t-\ell_1) = \mathcal{M}^s$  coefficients of  $F(X, Y)$  are chosen to be the data symbols that need to be stored on the DSS. Each node  $i \in [n]$  stores the evaluation of  $F(X, Y)$  as illustrated in (23). It follows from the description of the coding scheme of [21] in the beginning of this subsection that the resulting coding scheme is an exact repairable code at MBCR point.

Next, we show that the proposed scheme is indeed  $(\ell_1, 0)$ -secure. If  $\mathbf{e}$ ,  $\mathbf{u}$ , and  $\mathbf{r}$  denote the data observed by eavesdropper, original data to be stored, and the randomness added to the original data before encoding respectively, then it is sufficient to show (i)  $H(\mathbf{e}) \leq H(\mathbf{r})$  and (ii)  $H(\mathbf{r}|\mathbf{u}, \mathbf{e}) = 0$  in order to establish the secrecy claim (see Lemma 3). To argue the first requirement, noting that number of eavesdropped symbols are  $\ell_1\alpha = \ell_1(2d+t-1)$ , we will show that  $\ell_1^2 - \ell_1$  number of these are linearly dependent on the remaining ones. The eavesdropper, without loss of generality considering the first  $\ell_1$  nodes, observes the symbols given in Fig. 4. Due to the code construction, each row above represents evaluations of a polynomial of degree less than  $d+t$  and each column represents a polynomial of degree less than  $d$ . Hence, we observe that each of the symbols denoted with bold (blue) font in the matrix of Fig. 4 is a linear combination of the remaining ones. Therefore,  $H(\mathbf{e}) = \ell_1\alpha - \ell_1(\ell_1 - 1) = H(\mathbf{r})$ .

In order to show that second requirement also holds, we present a method to decode randomness  $\mathbf{r}$  given  $\mathbf{u}$  and data stored on any  $\ell_1$  nodes. Once we know the data symbols  $\mathbf{u}$ , we can remove the monomials associated to data symbols in  $F(X, Y)$  and the contribution of these monomials from the polynomial evaluations stored on DSS. Let  $\hat{F}(X, Y)$  denote the bivariate polynomial that we obtain by removing the data monomials:

$$\begin{aligned} \hat{F}(X, Y) = & \sum_{\substack{0 \leq i < \ell_1, \\ 0 \leq j < \ell_1}} a_{ij} X^i Y^j + \sum_{\substack{0 \leq i < \ell_1, \\ \ell_1 \leq j < k}} a_{ij} X^i Y^j \\ & + \sum_{\substack{\ell_1 \leq i < k, \\ 0 \leq j < \ell_1}} a_{ij} X^i Y^j + \sum_{\substack{0 \leq i < \ell_1, \\ k \leq j < d+t}} b_{ij} X^i Y^j \\ & + \sum_{\substack{k \leq i < d, \\ 0 \leq j < \ell_1}} c_{ij} X^i Y^j \end{aligned} \quad (25)$$

$\hat{F}(X, Y)$  can be rewritten as:

$$\begin{aligned} \hat{F}(X, Y) = & \sum_{\substack{0 \leq i < \ell_1, \\ 0 \leq j < \ell_1}} \hat{a}_{ij} X^i Y^j + \sum_{\substack{0 \leq i < \ell_1, \\ \ell_1 \leq j < d+t}} \hat{b}_{ij} X^i Y^j \\ & + \sum_{\substack{\ell_1 \leq i < d, \\ 0 \leq j < \ell_1}} \hat{c}_{ij} X^i Y^j \end{aligned} \quad (26)$$

where

$$\begin{aligned} \{\hat{a}_{ij}\}_{0 \leq i < \ell_1, 0 \leq j < \ell_1} &= \{a_{ij}\}_{0 \leq i < \ell_1, 0 \leq j < \ell_1}, \\ \{\hat{b}_{ij}\}_{0 \leq i < \ell_1, \ell_1 \leq j < k} &= \{a_{ij}\}_{0 \leq i < \ell_1, \ell_1 \leq j < k}, \\ \{\hat{b}_{ij}\}_{0 \leq i < \ell_1, k \leq j < d+t} &= \{b_{ij}\}_{0 \leq i < \ell_1, k \leq j < d+t}, \\ \{\hat{c}_{ij}\}_{\ell_1 \leq i < k, 0 \leq j < \ell_1} &= \{a_{ij}\}_{\ell_1 \leq i < k, 0 \leq j < \ell_1}, \\ \text{and } \{\hat{c}_{ij}\}_{k \leq i < d, 0 \leq j < \ell_1} &= \{c_{ij}\}_{k \leq i < d, 0 \leq j < \ell_1}. \end{aligned} \quad (27)$$

$\hat{F}(X, Y)$  in (26) takes the same form as  $F(X, Y)$  in (22) with  $k$  replaced with  $\ell_1$ . Therefore the randomness  $\mathbf{r}$ , coefficients of  $\hat{F}(X, Y)$  in (26), can be decoded from the data observed on  $\ell_1$  nodes using the data reconstruction method described in [21]. Thus, we obtain the following result.

**Proposition 6.** *The secrecy capacity at MBCR point for a file size of  $\mathcal{M}^s = k(2d - k + t)$  is given by  $\mathcal{M}^s = k(2d - k + t) - \ell_1(2d - \ell_1 + t)$  for any  $n \geq d + t$ .*

We list some instances of this construction in Table II in Appendix C. As evident from the table, cooperation helps to reduce the repair bandwidth if  $d < n - t$ . Thus, (secure) coding schemes for the case of  $n > d + t$  are of significant interest in order to reduce the repair bandwidth in cooperative repair.

#### IV. SECURE MSCR CODES

We first consider upper bound on the secure file size, and then utilize appropriate secrecy precoding mechanisms to construct achievable schemes.

##### A. Upper bound on the secure file size

At MSCR point, the nodes have minimum possible storage, i.e.,  $\alpha = \frac{\mathcal{M}}{k}$ . Using the cut-set analysis given in the previous section, one can obtain that the minimum repair bandwidth can be attained with  $\beta = \beta' = \frac{\alpha}{d-k+t} = \frac{\mathcal{M}}{k(d-k+t)}$ . (See also [13], [15].) At MSCR, therefore the downloaded data can be larger than the data stored in the nodes. Thus, for secrecy constraints, we consider two eavesdropper types: storage-only

eavesdroppers ( $\mathcal{E}_1$ ) and storage-and-download eavesdroppers ( $\mathcal{E}_2$ ). Using the size of these sets we denote the eavesdropper setting with  $(\ell_1, \ell_2)$  as introduced in Section II. Here, the eavesdroppers in  $\mathcal{E}_2$  observe both the downloaded data from live nodes and from that of cooperation nodes. Similar to the secure file size bound analysis given in the previous section, we obtain the following bound.

$$\mathcal{M}^s \leq \sum_{i=0}^{k-1} (1 - l_1^i - l_2^i) \min \left\{ \alpha - I(\mathbf{s}_i; \mathbf{d}_{i, \mathcal{E}_2}), (d-i)\beta + (t-1)\beta' \right\}. \quad (28)$$

Here, we consider  $u_i = 1$  number of nodes of stage  $i$  include  $l_1^i$  number of eavesdroppers from  $\mathcal{E}_1$  and  $l_2^i$  number of eavesdroppers from  $\mathcal{E}_2$ . Compared to the MSCR bounds, due to eavesdroppers in  $\mathcal{E}_2$ , nodes that are not eavesdropped may leak information during their participation in repair of a node having an  $\mathcal{E}_2$  type eavesdropper. Thus, the values of the cuts of type 1 include additional penalty terms  $I(\mathbf{s}_i; \mathbf{d}_{i, \mathcal{E}_2})$ , counting the leakage from the storage at the  $i$ -th node to nodes indexed with  $\mathcal{E}_2$ . (Here, the cut value can be written as  $H(\mathbf{s}_i | \mathbf{d}_{i, \mathcal{E}_2}) = H(\mathbf{s}_i) - I(\mathbf{s}_i; \mathbf{d}_{i, \mathcal{E}_2})$ .) Considering the MSCR point values of  $\alpha$ ,  $\beta$ , and  $\beta'$  given above, the second term of (28) will be loose. (The cases considered for (12) and (14), when specialized to the MSCR point, do not give tighter bound than that of (28).) Hence, considering that the first  $k - \ell_1 - \ell_2$  repairs are eavesdropper-free, (28) will evaluate to the following bound.

**Proposition 7.** *Cooperative regenerating codes operating at the MSCR point with a secure file size of  $\mathcal{M}^s$  satisfy*

$$\mathcal{M}^s \leq \sum_{i=0}^{k-\ell_1-\ell_2-1} \alpha - I(\mathbf{s}_i; \mathbf{d}_{i, \mathcal{E}_2}), \quad (29)$$

where MSCR point is given by  $\beta = \beta' = 1$ ,  $\alpha = d - k + t$ , for a file size of  $\mathcal{M} = k(d - k + t)$ . In addition, at MSCR, one can bound  $I(\mathbf{s}_i; \mathbf{d}_{i, \mathcal{E}_2}) \geq \beta' = \beta$  and obtain the bound

$$\mathcal{M}^s \leq (k - \ell_1 - \ell_2)(\alpha - \beta).$$

#### B. Code construction for secure MSCR when $k = t = 2$

We consider an interference alignment approach based on the one proposed in [19], considering  $k = t = 2$ . (See also [48], [49].) For any  $(n, k, d, t)$  with  $d \geq k$  and  $n = d + t$ , we have  $\alpha = d - k + t = n - 2$ , and  $\mathcal{M} = k(d - k + t) = 2(d - k + t) = 2\alpha$  at MSCR point. From the bound given in Proposition 7, the achievability of positive secure file size is possible only when  $(\ell_1, \ell_2) = (1, 0)$  or  $(\ell_1, \ell_2) = (0, 1)$  when  $k = 2$ . Corresponding bounds are given by  $\mathcal{M}^s \leq \alpha$  and  $\mathcal{M}^s \leq \alpha - 1$ , respectively. (For the latter bound, as  $|\mathcal{E}_2| = 1$ ,  $\mathbf{d}_{i, \mathcal{E}_2}$  necessarily consists of one symbol as  $\beta = \beta' = 1$ , and the non-eavesdropped node participates in the repair of the eavesdropped node by sending  $\beta$  or  $\beta'$  symbols.) In the following, we construct codes achieving the stated bounds for both cases, hence establishing the secrecy capacity when  $k = t = 2$ . We show this with codes having  $n = d + t$ , i.e., all the nodes participate in the repair. The construction can

be extended to cases with  $n > d + t$  by following a similar approach and choosing a larger field size.

**Case 1:**  $\mathcal{M}^s = \alpha$  **when**  $(\ell_1, \ell_2) = (1, 0)$

Consider a finite field size of  $q = n - 1$  with generator  $w$ ,  $\alpha$  number of random symbols  $r_1, \dots, r_\alpha$ , and  $\alpha$  number of secure information symbols  $s_1, \dots, s_\alpha$ . Both information and random symbols are uniformly distributed over the field. We construct the file given by  $\mathcal{M} = \{a_1 \triangleq r_1, \dots, a_\alpha \triangleq r_\alpha, b_1 \triangleq r_1 + s_1, \dots, b_\alpha \triangleq r_\alpha + s_\alpha\}$ , and consider the following placement

- Store  $\mathbf{a} = (a_1, \dots, a_\alpha)^T$  at the first node,
- Store  $\mathbf{b} = (b_1, \dots, b_\alpha)^T$  at the second node, and
- Store  $\mathbf{r}_i = (a_1 + w^{(i-1) \bmod \alpha} b_1, \dots, a_\alpha + w^{(i+\alpha-2) \bmod \alpha} b_\alpha)^T$  at the  $i$ -th redundancy node,  $i \in \{1, \dots, \alpha\}$ .

Data collector can reconstruct the file  $\mathcal{M}$  by contacting any of the  $k$  nodes, and solving  $\alpha$  groups of 2 equations over 2 unknowns for each group. From file  $\mathcal{M}$ , it can then obtain the secure symbols  $s_1, \dots, s_\alpha$ . For cooperative repair, considering the repair of the first systematic node,  $i$ -th redundancy node storing  $\mathbf{r}_i = \mathbf{a} + \mathbf{B}_i \mathbf{b}$ , sends  $\mathbf{v}_{1,i} \mathbf{r}_i = \mathbf{w}_{1,i} \mathbf{a} + \mathbf{z} \mathbf{b}$ , where  $\mathbf{v}_{1,i} = \mathbf{z} \mathbf{B}_i^{-1}$  and  $\mathbf{z} = (1, \dots, 1)$ . (Repair of the second systematic node is symmetric to the first one, and, without loss of generality, we consider the repair of the two systematic nodes. Repairs of stages involving redundancy nodes can be performed as that of the systematic nodes after change of variables.) Second systematic node, having received  $\mathbf{c}_2 = \{\mathbf{v}_{2,1} \mathbf{r}_1, \dots, \mathbf{v}_{2,d} \mathbf{r}_d\}$ , chooses the repair vector  $\mathbf{v}_{1,0}$  such that  $\mathbf{v}_{1,0} \mathbf{c}_2 = \mathbf{w}_{1,0} \mathbf{a} + \mathbf{z} \mathbf{b}$ ; and sends  $\mathbf{v}_{1,0} \mathbf{c}_2$  to the first systematic node. Then, the first systematic node solves  $d + 1$  equations  $\{\mathbf{w}_{1,0} \mathbf{a} + \mathbf{z} \mathbf{b}, \mathbf{w}_{1,1} \mathbf{a} + \mathbf{z} \mathbf{b}, \dots, \mathbf{w}_{1,d} \mathbf{a} + \mathbf{z} \mathbf{b}\}$  in  $d + 1$  unknowns  $\{a_1, \dots, a_\alpha, \mathbf{z} \mathbf{b}\}$ . Noting that the regeneration and repair are similar to the ones proposed in [19], it remains to show the secrecy of the file. Here, regardless of eavesdropped node being in the systematic or parity nodes, given the secure symbols,  $\mathbf{u} = s_1, \dots, s_\alpha$ , the eavesdropper can obtain  $\alpha$  equations in  $\alpha$  unknowns  $\mathbf{r} = r_1, \dots, r_\alpha$ ; and solve for  $\mathbf{r}$ . This shows that  $H(\mathbf{r} | \mathbf{u}, \mathbf{e}) = 0$ , where the eavesdropper observes the content of the eavesdropped node, i.e.,  $\mathbf{e} = \mathbf{s}_{\mathcal{E}_1}$ . We see that, at the eavesdropped node, the content of the stored data necessarily satisfies  $H(\mathbf{e}) = H(\mathbf{s}_{\mathcal{E}_1}) = \alpha$ . Then, as the code satisfies both  $H(\mathbf{e}) \leq H(\mathbf{r})$  and  $H(\mathbf{r} | \mathbf{u}, \mathbf{e}) = 0$ , we obtain from Lemma 3 that  $I(\mathbf{u}; \mathbf{e}) = I(s_1, \dots, s_\alpha; \mathbf{s}_{\mathcal{E}_1}) = 0$ .

**Case 2:**  $\mathcal{M}^s = \alpha - 1$  **when**  $(\ell_1, \ell_2) = (0, 1)$

We modify the above construction by considering the file given by  $\mathcal{M} = \{a_1 \triangleq r_1, \dots, a_\alpha \triangleq r_\alpha, b_1 \triangleq r_1 + s_1, \dots, b_{\alpha-1} \triangleq r_{\alpha-1} + s_{\alpha-1}, b_\alpha \triangleq r_\alpha + s_\alpha\}$ . The regeneration and repair parts are the same as that of the previous section. We show that the secrecy constraint is satisfied here. The content of the eavesdropped node  $\mathbf{s}_{\mathcal{E}_2}$  is generated from the downloaded data  $\mathbf{d}_{\mathcal{E}_2}$ . Thus, we need to show  $I(\mathbf{u}; \mathbf{e}) = 0$  with  $\mathbf{u} = \{s_1, \dots, s_{\alpha-1}\}$  and  $\mathbf{e} = \mathbf{d}_{\mathcal{E}_2}$ . Without loss generality, we consider the eavesdropper observing the first systematic node. Considering the repair process described above, we have  $\mathbf{e} = \mathbf{d}_{\mathcal{E}_2} = \{\mathbf{w}_{1,0} \mathbf{a} + \mathbf{z} \mathbf{b}, \mathbf{w}_{1,1} \mathbf{a} + \mathbf{z} \mathbf{b}, \dots, \mathbf{w}_{1,d} \mathbf{a} + \mathbf{z} \mathbf{b}\}$ , from which we obtain that  $H(\mathbf{e}) \leq \alpha + 1$ . In addition, as the eavesdropper can solve for  $(\mathbf{a}, \mathbf{z} \mathbf{b})$ , it can solve for  $\mathbf{r} = \{r_1, \dots, r_{\alpha+1}\}$  from the  $\alpha + 1$  number of equations in  $(\mathbf{a}, \mathbf{z} \mathbf{b})$ , after canceling

out the secure symbols  $\mathbf{u} = \{s_1, \dots, s_{\alpha-1}\}$ . This shows that  $H(\mathbf{r}|\mathbf{u}, \mathbf{e}) = 0$ . This, together with  $H(\mathbf{r}) = \alpha + 1$  and Lemma 3, we obtain that  $I(\mathbf{u}; \mathbf{e}) = I(s_1, \dots, s_{\alpha-1}; \mathbf{d}_{\mathcal{E}_2}) = 0$ .

**Proposition 8.** *The secrecy capacity at MSCR point for a file size of  $\mathcal{M} = k(d - k + t)$  is given by  $\mathcal{M}^s = \alpha$ , if  $(\ell_1, \ell_2) = (1, 0)$  and  $k = t = 2$ ; and by  $\mathcal{M}^s = \alpha - 1$ , if  $(\ell_1, \ell_2) = (0, 1)$  and  $k = t = 2$ .*

### C. Code construction for secure MSCR when $d = k$

The above construction is limited to the  $k = 2$  case. Here, we provide secure MSCR code when  $d = k$ , and hence allowing  $k > 2$ . (Note that as  $d \geq k \geq \ell_1 + \ell_2$ , we necessarily have  $\ell_1 + \ell_2 \leq d = k$  here.) Again, we apply the two-stage encoding, with using an MRD code as the secrecy pre-coding.

Consider  $\mathcal{M} = k(d - k + t) = kt$ ,  $\beta = \beta' = 1$ ,  $\alpha = d - k + t = t$ ,  $\mathcal{M}^s = kt - (\ell_1 + \ell_2)t - \ell_2(k - \ell_1 - \ell_2)$ , and  $n \geq d + t$ . We encode the data using the linearized polynomial  $f(g) = \sum_{i=0}^{\mathcal{M}-1} u_i g^{q^i}$ . (This is the Gabidulin construction of MRD codes [44] summarized in Section III-B.) The coefficients of  $f(g)$  is chosen by  $\mathcal{M}^s$  data symbols denoted by  $\mathbf{u}$  and  $\mathcal{M} - \mathcal{M}^s$  random symbols denoted by  $\mathbf{r}$ . The function  $f(g)$  is evaluated at  $\mathcal{M}$  points in  $\mathbb{F}_{q^m} \{g_1, \dots, g_{\mathcal{M}}\}$  that are linearly independent over  $\mathbb{F}_q$ . (Here, the data and random symbols belong to  $\mathbb{F}_{q^m}$  with  $m \geq \mathcal{M}$ .) We denote these points as  $x_i = f(\alpha_i)$  for  $i = 1, \dots, \mathcal{M} = kt$ . We consider the code provided in [17] in the secrecy setting here. We place  $\mathcal{M} = kt$  symbols into vectors  $\mathbf{m}_1, \dots, \mathbf{m}_t$ , each having  $k$  symbols. We encode these vectors with a Vandermonde matrix of size  $k \times n$ , whose columns are represented as  $\mathbf{g}_i$  for  $i = 1, \dots, n$ . We store  $\mathbf{m}_j^T \mathbf{g}_i$  at node  $i$ . Data collector, by contacting any  $k$  nodes, can obtain  $k$  equations for each of  $\mathbf{m}_j$ , and solve them to obtain  $x_i$  for  $i = 1, \dots, \mathcal{M} = kt$ . It can then obtain the secure data symbols by interpolation. For node repair, consider that node  $j \in [t]$  contacts  $d = k$  live nodes, named as  $j_1$  to  $j_k$ . It will download  $\mathbf{m}_j^T \mathbf{g}_{j_l}$  from live node  $j_l$  for  $l = 1, \dots, k$ . Node  $j$  then will obtain  $\mathbf{m}_j$  by solving these  $k$  equations, and send  $\mathbf{m}_j \mathbf{g}_{j'}$  to  $j' \neq j$ ,  $j' \in [t]$ , the remaining nodes under repair. Each node  $j \in [t]$  will repeat this procedure. (Then, node  $j$  will also recover its  $\mathbf{m}_{j'} \mathbf{g}_j$  by downloading a symbol from each of the node being under repair.)

We here show the secrecy constraint has met assuming  $\ell_2 \leq t$ . (Otherwise, this construction can not achieve a positive secure file size as the  $\mathcal{E}_2$  eavesdroppers can obtain all  $\mathbf{m}_{[1:t]}$  symbols from their downloads.) We observe that  $\mathcal{E}_2$  nodes being under repair obtain  $\ell_2 k$  equations from the live nodes (these will reveal  $\ell_2$  number of  $\mathbf{m}_j \mathbf{s}$ ), and store additional  $\ell_2(t - \ell_2) = \ell_2(\alpha - \ell_2)$  symbols received from the remaining nodes under repair.  $\mathcal{E}_1$  nodes observe  $\ell_1 \alpha$  number of symbols. However,  $\ell_1 \ell_2$  of these symbols are linearly dependent to the ones downloaded by the  $\mathcal{E}_2$  nodes (as  $\mathcal{E}_2$  nodes have the knowledge of  $\ell_2$  number of  $\mathbf{m}_j \mathbf{s}$ ). Therefore, using the given polynomial and the secure data of length  $\mathcal{M}^s$ , the eavesdropper can solve for the random symbols using these  $\ell_2(k + \alpha - \ell_2) + \ell_1(\alpha - \ell_2) = \ell_2(k + t - \ell_2) + \ell_1(t - \ell_2) = (k - \ell_1 - \ell_2)\ell_2 + (\ell_1 + \ell_2)t = \mathcal{M} - \mathcal{M}^s$  linearly independent evaluations of the polynomial. Thus, we have  $H(\mathbf{r}|\mathbf{u}, \mathbf{e}) = 0$ ,

where  $\mathbf{e}$  denotes the observations of  $\mathcal{E}_1$  and  $\mathcal{E}_2$  eavesdroppers. This construction also satisfies  $H(\mathbf{e}) = \ell_2 k + (\alpha - \ell_2)\ell_2 + \ell_1(\alpha - \ell_2) = H(\mathbf{r})$  as argued above, and it follows from Lemma 3 that we have  $I(\mathbf{u}; \mathbf{e}) = 0$ . This code achieves the secure file size of  $kt - (\ell_1 + \ell_2)t - \ell_2(k - \ell_1 - \ell_2)$  when  $\ell_2 \leq t$ .

**Proposition 9.** *The secure file size of  $\mathcal{M}^s = (k - \ell_1 - \ell_2)[t - \ell_2]^+$  is achievable at the MSCR point for a file size of  $\mathcal{M} = k(d - k + t)$  when  $d = k$ .*

Note that this achieves the secrecy capacity when  $\ell_2 \leq 1$  for any  $\ell_1$  as can be observed from the bound given by Proposition 7.

## V. CONCLUSION

DSS store data in multiple nodes. These systems not only require resilience against node failures, but also have to satisfy security constraints and to perform multiple node repair. Regenerating codes proposed for DSS address the node failure resilience while efficiently trading off storage vs. repair bandwidth. In this paper, we considered secure cooperative regenerating codes for DSS. The eavesdropper model analyzed in this paper belongs to the class of passive attack models, where the eavesdroppers observe the content of the nodes in the system. Accordingly, we considered an  $(\ell_1, \ell_2)$ -eavesdropper, where the storage content of any  $\ell_1$  nodes, and the download content of any  $\ell_2$  nodes are leaked to the eavesdropper. With such an eavesdropper model, we studied the security for the multiple repair scenario, in particular secure cooperative regenerating codes. For the minimum bandwidth cooperative regenerating (MBCR) point, we established a bound on the secrecy capacity, and by modifying the existing coding schemes in the literature, devised new codes achieving the secrecy capacity. For the minimum storage cooperative regenerating (MSCR) point, on the other hand, we proposed an upper bound and lower bounds on the secure file size, which match under special cases. The results show that it is possible to design regenerating codes that not only efficiently trades storage vs. repair bandwidth, but also resilient against security attacks in a cooperative repair scenario. Finally, as evident from some of our secrecy-achieving constructions, we would like to emphasize the role that the maximum rank distance (MRD) codes can take in secrecy problems. In particular, we have utilized the Gabidulin construction [44] of MRD codes and properties of linearized polynomials in obtaining some of the results. Similar properties of such codes have been utilized to achieve secrecy in earlier works [50]–[53], and they proved their potential again here as an essential component for achieving secrecy in DSS.

We list some avenues for further research here. The secrecy capacity of MSCR codes remain as an open problem, as we have established the optimal codes under some parameter settings. To attempt this problem, codes for MSCR without security constraints have to be further investigated. One can also consider cooperative repair in a DSS having locally repairable structure. As other distributed systems, DSS may exhibit simultaneous node failures that need to be recovered with local connections. According to our best knowledge,

this setting has not been studied (even without security constraints). Our ongoing efforts are on the design of coding schemes for DSS satisfying these properties.

#### APPENDIX A PROOF OF LEMMA 3

*Proof:* The proof follows from the classical techniques given by [25], where instead of 0-leakage,  $\epsilon$ -leakage rate is considered. (The application of this technique in DSS is also considered in [27].) We have

$$I(\mathbf{u}; \mathbf{e}) = H(\mathbf{e}) - H(\mathbf{e}|\mathbf{u}) \quad (30)$$

$$\stackrel{(a)}{\leq} H(\mathbf{e}) - H(\mathbf{e}|\mathbf{u}) + H(\mathbf{e}|\mathbf{u}, \mathbf{r}) \quad (31)$$

$$\stackrel{(b)}{\leq} H(\mathbf{r}) - I(\mathbf{e}; \mathbf{r}|\mathbf{u}) \quad (32)$$

$$\stackrel{(c)}{=} H(\mathbf{r}|\mathbf{u}, \mathbf{e}) \quad (33)$$

$$\stackrel{(d)}{=} 0 \quad (34)$$

where (a) follows by non-negativity of  $H(\mathbf{e}|\mathbf{u}, \mathbf{r})$ , (b) is the condition  $H(\mathbf{e}) \leq H(\mathbf{r})$ , (c) is due to  $H(\mathbf{r}|\mathbf{u}) = H(\mathbf{r})$  as  $\mathbf{r}$  and  $\mathbf{u}$  are independent, (d) is the condition  $H(\mathbf{r}|\mathbf{u}, \mathbf{e}) = 0$ .

**Remark 10.** *If the eavesdropper has a vanishing probability of error in decoding  $\mathbf{r}$  given  $\mathbf{e}$  and  $\mathbf{u}$ , then, by Fano's inequality, one can write  $H(\mathbf{r}|\mathbf{u}, \mathbf{e}) \leq |\mathbf{r}|\epsilon$ , and, by following the above steps, can show the bound  $I(\mathbf{u}; \mathbf{e}) \leq |\mathbf{r}|\epsilon$ , where  $|\mathbf{r}|$  is the number of random bits, and  $\epsilon$  can be made small if the probability of error is vanishing. This shows that the leakage rate  $I(\mathbf{u}; \mathbf{e})/|\mathbf{e}|$  is vanishing. (See, e.g., [25].)*

#### APPENDIX B PROOF OF PROPOSITION 4

*Proof:* (11) evaluates to the following bound at the MBCR point for a file size of  $\mathcal{M} = k(2d - k + t)$ .

$$\begin{aligned} \mathcal{M}^s &\leq \bar{\mathcal{M}}^s \triangleq k(2d - k + t) - \ell_1(2d - \ell_1 + t) \\ &= (k - \ell_1)(2d + t - k - \ell_1) \end{aligned} \quad (35)$$

We compare this with the bounds (12) and (14).

- If  $t \geq k$ : (12) evaluates to the following bound

$$\mathcal{M}^s \leq \bar{\mathcal{M}}_1^s \triangleq (k - \ell_1)(2d + t - k). \quad (36)$$

Here, as  $\ell_1 \geq 0$ ,  $\bar{\mathcal{M}}^s \leq \bar{\mathcal{M}}_1^s$ . Hence, (35) gives a tighter bound.

- If  $t \leq k$  and  $\ell_1 \leq at$  with  $a = \lfloor k/t \rfloor$ : Let  $\ell_1 = \tilde{a}t + \tilde{b}$ , where  $\tilde{a} = \lfloor \ell_1/t \rfloor$  and  $\tilde{b} \in [0, t - 1]$ . The expression of  $S$  in (14) at MBCR point is given by

$$\begin{aligned} S &= 2\ell_1(d - \tilde{a}t) + t^2\tilde{a}(\tilde{a} + 1) \\ &\stackrel{(a)}{=} \ell_1(2d - 2\ell_1 + 2\tilde{b}) + (\ell_1 - \tilde{b})(\ell_1 - \tilde{b} + t) \\ &= \ell_1(2d - \ell_1 + \tilde{b} + t) - \tilde{b}(\ell_1 - \tilde{b} + t) \\ &= \ell_1(2d - \ell_1 + t) - \tilde{b}(t - \tilde{b}) \end{aligned} \quad (37)$$

where in (a) we used  $\tilde{a}t = \ell_1 - \tilde{b}$ . Therefore, (14) evaluates to

$$\begin{aligned} \mathcal{M}^s &\leq \bar{\mathcal{M}}_2^s \triangleq k(2d + t - k) - \ell_1(2d - \ell_1 + t) \\ &\quad + \tilde{b}(t - \tilde{b}). \end{aligned} \quad (38)$$

As  $\tilde{b}(t - \tilde{b}) \geq 0$ , we obtain that  $\bar{\mathcal{M}}^s \leq \bar{\mathcal{M}}_2^s$ . Hence, (35) gives a tighter bound.

- If  $t \leq k$  and  $\ell_1 \geq at$  with  $a = \lfloor k/t \rfloor$ : Let  $\ell_1 = at + \tilde{b}$ , where  $a = \lfloor \ell_1/t \rfloor$  and  $\tilde{b} \in [0, t - 1]$ . The expression of  $S$  in (14) at MBCR point is given by

$$\begin{aligned} S &= 2\ell_1(d - at) + t^2a(a + 1) + (\ell_1 - at)(t - b) \\ &\stackrel{(a)}{=} \ell_1(2d - 2\ell_1 + 2\tilde{b}) + (\ell_1 - \tilde{b})(\ell_1 - \tilde{b} + t) \\ &\quad + \tilde{b}(t - b) \\ &= \ell_1(2d - \ell_1 + t) - \tilde{b}(b - \tilde{b}) \end{aligned} \quad (39)$$

where in (a) we used  $at = \ell_1 - \tilde{b}$ . Therefore, (14) evaluates to

$$\begin{aligned} \mathcal{M}^s &\leq \bar{\mathcal{M}}_3^s \triangleq k(2d + t - k) - \ell_1(2d - \ell_1 + t) \\ &\quad + \tilde{b}(b - \tilde{b}). \end{aligned} \quad (40)$$

As  $\tilde{b}(b - \tilde{b}) \geq 0$  due to  $k \geq \ell_1$ , we obtain that  $\bar{\mathcal{M}}^s \leq \bar{\mathcal{M}}_3^s$ . Hence, (35) gives a tighter bound.

Combining the cases above, we see that the upper bound on the secure MBCR file size is given by (35). ■

#### APPENDIX C NRBW VALUES FOR MBCR POINT IN DSS

The parameters of Proposition 4 are given in the following tables.  $\ell_1 = 0$  case corresponds to the systems without security constraints.  $t = 1$  case corresponds to non-cooperative case. Red (green) font highlights cases with greater (respectively, smaller) cooperative NRBW ( $\gamma/\mathcal{M}^s$ ) compared that of  $t = 1$ . We observed that the same trend continues for higher  $n$  values.

TABLE I: NRBW for  $n = 4, 5$ ,  $d \geq k$ ,  $d + t = n$ .

$n$	$k$	$l$	$t$	$d$	$\beta/\mathcal{M}^s$	$\beta'/\mathcal{M}^s$	$\gamma/\mathcal{M}^s$	$\mathcal{M}$	$\mathcal{M}^s$
4	2	0	1	3	0.2000	0.1000	0.6000	10	10
4	2	0	2	2	0.2500	0.1250	<b>0.6250</b>	8	8
4	2	1	1	3	0.5000	0.2500	1.5000	10	4
4	2	1	2	2	0.6667	0.3333	<b>1.6667</b>	8	3
4	3	0	1	3	0.1667	0.0833	0.5000	12	12
4	3	1	1	3	0.3333	0.1667	1.0000	12	6
4	3	2	1	3	1.0000	0.5000	3.0000	12	2
5	2	0	1	4	0.1429	0.0714	0.5714	14	14
5	2	0	2	3	0.1667	0.0833	<b>0.5833</b>	12	12
5	2	0	3	2	0.2000	0.1000	<b>0.6000</b>	10	10
5	2	1	1	4	0.3333	0.1667	1.3333	14	6
5	2	1	2	3	0.4000	0.2000	<b>1.4000</b>	12	5
5	2	1	3	2	0.5000	0.2500	<b>1.5000</b>	10	4
5	3	0	1	4	0.1111	0.0556	0.4444	18	18
5	3	0	2	3	0.1333	0.0667	<b>0.4667</b>	15	15
5	3	1	1	4	0.2000	0.1000	0.8000	18	10
5	3	1	2	3	0.2500	0.1250	<b>0.8750</b>	15	8
5	3	2	1	4	0.5000	0.2500	2.0000	18	4
5	3	2	2	3	0.6667	0.3333	<b>2.3333</b>	15	3
5	4	0	1	4	0.1000	0.0500	0.4000	20	20
5	4	1	1	4	0.1667	0.0833	0.6667	20	12
5	4	2	1	4	0.3333	0.1667	1.3333	20	6
5	4	3	1	4	1.0000	0.5000	4.0000	20	2



TABLE II: NRBW for  $n = 4, 5$ ,  $d \geq k$ ,  $d + t \leq n$ .

$n$	$k$	$l$	$t$	$d$	$\beta/\mathcal{M}^s$	$\beta'/\mathcal{M}^s$	$\gamma/\mathcal{M}^s$	$\mathcal{M}$	$\mathcal{M}^s$
4	2	0	1	3	0.2000	0.1000	0.6000	10	10
4	2	0	1	2	0.3333	0.1667	0.6667	6	6
4	2	0	2	2	0.2500	0.1250	0.6250	8	8
4	2	1	1	3	0.5000	0.2500	1.5000	10	4
4	2	1	1	2	1.0000	0.5000	2.0000	6	2
4	2	1	2	2	0.6667	0.3333	1.6667	8	3
4	3	0	1	3	0.1667	0.0833	0.5000	12	12
4	3	1	1	3	0.3333	0.1667	1.0000	12	6
4	3	2	1	3	1.0000	0.5000	3.0000	12	2
5	2	0	1	4	0.1429	0.0714	0.5714	14	14
5	2	0	1	3	0.2000	0.1000	0.6000	10	10
5	2	0	2	3	0.1667	0.0833	0.5833	12	12
5	2	0	1	2	0.3333	0.1667	0.6667	6	6
5	2	0	2	2	0.2500	0.1250	0.6250	8	8
5	2	0	3	2	0.2000	0.1000	0.6000	10	10
5	2	1	1	4	0.3333	0.1667	1.3333	14	6
5	2	1	1	3	0.5000	0.2500	1.5000	10	4
5	2	1	2	3	0.4000	0.2000	1.4000	12	5
5	2	1	1	2	1.0000	0.5000	2.0000	6	2
5	2	1	2	2	0.6667	0.3333	1.6667	8	3
5	2	1	3	2	0.5000	0.2500	1.5000	10	4
5	3	0	1	4	0.1111	0.0556	0.4444	18	18
5	3	0	1	3	0.1667	0.0833	0.5000	12	12
5	3	0	2	3	0.1333	0.0667	0.4667	15	15
5	3	1	1	4	0.2000	0.1000	0.8000	18	10
5	3	1	1	3	0.3333	0.1667	1.0000	12	6
5	3	1	2	3	0.2500	0.1250	0.8750	15	8
5	3	2	1	4	0.5000	0.2500	2.0000	18	4
5	3	2	1	3	1.0000	0.5000	3.0000	12	2
5	3	2	2	3	0.6667	0.3333	2.3333	15	3
5	4	0	1	4	0.1000	0.0500	0.4000	20	20
5	4	1	1	4	0.1667	0.0833	0.6667	20	12
5	4	2	1	4	0.3333	0.1667	1.3333	20	6
5	4	3	1	4	1.0000	0.5000	4.0000	20	2

## REFERENCES

- [1] S. Rhea, P. Eaton, D. Geels, H. Weatherspoon, B. Zhao, and J. Kubiatowicz, "Pond: the OceanStore Prototype," in *Proc. of the 2nd USENIX Conference on File and Storage Technologies (FAST 03)*, 2003.
- [2] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The Google file system," in *Proceedings of the nineteenth ACM symposium on Operating systems principles*, ser. SOSP '03. New York, NY, USA: ACM, 2003, pp. 29–43. [Online]. Available: <http://doi.acm.org/10.1145/945445.945450>
- [3] R. Bhagwan, K. Tati, Y.-C. Cheng, S. Savage, and G. M. Voelker, "Total recall: system support for automated availability management," in *Proceedings of the 1st conference on Symposium on Networked Systems Design and Implementation - Volume 1*, ser. NSDI'04. Berkeley, CA, USA: USENIX Association, 2004, pp. 25–25. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1251175>
- [4] A. Dimakis, P. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4539–4551, Sep. 2010.
- [5] S. Pawar, S. El Rouayheb, and K. Ramchandran, "Securing dynamic distributed storage systems against eavesdropping and adversarial attacks," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6734–6753, Oct. 2011.
- [6] K. Rashmi, N. Shah, and P. Kumar, "Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5227–5239, Aug. 2011.
- [7] D. Papailiopoulos, A. Dimakis, and V. Cadambe, "Repair optimal erasure codes through hadamard designs," in *Proc. 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton 2011)*, Sep. 2011, pp. 1382–1389.
- [8] V. Cadambe, C. Huang, and J. Li, "Permutation code: Optimal exact-repair of a single failed node in MDS code based distributed storage systems," in *Proc. 2011 IEEE International Symposium on Information Theory (ISIT 2011)*, Jul. 31-Aug. 5 2011, pp. 1225–1229.
- [9] I. Tamo, Z. Wang, and J. Bruck, "Zigzag codes: MDS array codes with optimal rebuilding," *CoRR*, vol. abs/1112.0371, Dec. 2011.
- [10] Z. Wang, I. Tamo, and J. Bruck, "On codes for optimal rebuilding access," in *Proc. 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton 2011)*, Sep. 2011, pp. 1374–1381.
- [11] Y. Hu, Y. Xu, X. Wang, C. Zhan, and P. Li, "Cooperative recovery of distributed storage systems from multiple losses with network coding," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 2, pp. 268–276, Feb. 2010.
- [12] X. Wang, Y. Xu, Y. Hu, and K. Ou, "MFR: Multi-loss flexible recovery in distributed storage systems," in *Proc. 2010 IEEE International Conference on Communications (ICC 2010)*, May 2010.
- [13] A.-M. Kermarrec, N. Le Scouarnec, and G. Straub, "Repairing multiple failures with coordinated and adaptive regenerating codes," in *Proc. 2011 International Symposium on Network Coding (NetCod 2011)*, Jul. 2011.
- [14] K. Shum and Y. Hu, "Existence of minimum-repair-bandwidth cooperative regenerating codes," in *Proc. 2011 International Symposium on Network Coding (NetCod 2011)*, Jul. 2011.
- [15] F. Oggier, "Coding techniques for distributed storage systems," NTU, Tech. Rep., Mar. 2012.
- [16] K.-W. Shum and Y. Hu, "Exact minimum-repair-bandwidth cooperative regenerating codes for distributed storage systems," in *Proc. 2011 IEEE International Symposium on Information Theory Proceedings (ISIT 2011)*, Jul. 31-Aug. 5 2011, pp. 1442–1446.
- [17] K. W. Shum, "Cooperative regenerating codes for distributed storage systems," in *Proc. 2011 IEEE International Conference on Communications (ICC 2011)*, Jun. 2011.
- [18] K. W. Shum and Y. Hu, "Cooperative regenerating codes," *CoRR*, vol. abs/1207.6762, Jul. 2012.
- [19] N. Le Scouarnec, "Exact scalar minimum storage coordinated regenerating codes," in *Proc. 2012 IEEE International Symposium on Information Theory Proceedings (ISIT 2012)*, Jul. 2012.
- [20] S. Jieka and N. L. Scouarnec, "CROSS-MBCR: Exact minimum bandwidth coordinated regenerating codes," *CoRR*, vol. abs/1207.0854, Jul. 2012.
- [21] A. Wang and Z. Zhang, "Exact cooperative regenerating codes with minimum-repair-bandwidth for distributed storage," *CoRR*, vol. abs/1207.0879, Jul. 2012.
- [22] O. Goldreich, *Foundations of Cryptography: Volume II, Basic Applications*. Cambridge University Press, 2004.
- [23] H. Delfs and H. Knebl, *Introduction to Cryptography: Principles and Applications*, 2nd ed. Springer, 2007.
- [24] C. E. Shannon, "Communication theory of secrecy systems," *The Bell System Technical Journal*, vol. 28, pp. 656–715, 1949.
- [25] A. Wyner, "The wire-tap channel," *The Bell System Technical Journal*, vol. 54, no. 8, pp. 1355–1387, Oct. 1975.
- [26] G. S. Vernam, "Cipher printing telegraph systems for secret wire and radio telegraphic communications," *J. Amer. Inst. Elect. Eng.*, vol. 55, pp. 109–115, 1926.
- [27] N. B. Shah, K. V. Rashmi, and P. V. Kumar, "Information-theoretically secure regenerating codes for distributed storage," in *Proc. 2011 IEEE Global Telecommunications Conference (GLOBECOM 2011)*, Dec. 2011.
- [28] F. E. Oggier and A. Datta, "Homomorphic self-repairing codes for agile maintenance of distributed storage systems," *CoRR*, vol. abs/1107.3129, Jul. 2011.
- [29] —, "Self-Repairing codes for distributed storage - A projective geometric construction," *CoRR*, vol. abs/1105.0379, May 2011.
- [30] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the locality of codeword symbols," *CoRR*, vol. abs/1106.3625, Jun. 2011.
- [31] C. Huang, M. Chen, and J. Li, "Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems," in *Proc. Sixth IEEE International Symposium on Network Computing and Applications (NCA 2007)*, Jul. 2007.
- [32] N. Prakash, G. Kamath, V. Lalitha, and P. Kumar, "Optimal linear codes with a local-error-correction property," in *2012 IEEE International Symposium on Information Theory Proceedings (ISIT 2012)*, Jul. 2012.
- [33] D. Papailiopoulos and A. Dimakis, "Locally repairable codes," in *2012 IEEE International Symposium on Information Theory Proceedings (ISIT 2012)*, Jul. 2012.
- [34] A. S. Rawat, O. O. Koyluoglu, N. Silberstein, and S. Vishwanath, "Optimal locally repairable and secure codes for distributed storage systems," 2012, in preparation.
- [35] F. Oggier and A. Datta, "Byzantine fault tolerance of regenerating codes," in *Proc. 2011 IEEE International Conference on Peer-to-Peer Computing (P2P)*, Aug. 31 - Sep. 2, 2011.
- [36] K. Rashmi, N. Shah, K. Ramchandran, and P. Kumar, "Regenerating codes for errors and erasures in distributed storage," in *2012 IEEE*

- International Symposium on Information Theory Proceedings (ISIT 2012)*, Jul. 2012.
- [37] N. Silberstein, A. S. Rawat, and S. Vishwanath, "Error resilience in distributed storage via rank-metric codes," in *Proc. 50th Allerton Conference on Communication, Control, and Computing (Allerton 2012)*, 2012.
  - [38] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979. [Online]. Available: <http://dl.acm.org/citation.cfm?id=359176>
  - [39] D. A. Patterson, G. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks (RAID)," in *Proceedings of the 1988 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '88. New York, NY, USA: ACM, 1988, pp. 109–116. [Online]. Available: <http://doi.acm.org/10.1145/50202.50214>
  - [40] M. Blaum, J. Brady, J. Bruck, and J. Menon, "EVENODD: An efficient scheme for tolerating double disk failures in RAID architectures," *IEEE Transactions on Computers*, vol. 44, no. 2, pp. 192–202, Feb. 1995.
  - [41] M. Blaum, J. Bruck, and A. Vardy, "MDS array codes with independent parity symbols," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 529–542, Mar. 1996.
  - [42] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.
  - [43] T. Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.
  - [44] É. M. Gabidulin, "Theory of codes with maximum rank distance," *Probl. Peredachi Inf. (Problems of Information Transmission)*, vol. 21, no. 1, pp. 3–16, Jul. 1985.
  - [45] P. Delsarte, "Bilinear forms over a finite field, with applications to coding theory," *Journal of Combinatorial Theory, Series A*, vol. 25, no. 3, pp. 226 – 241, 1978. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0097316578900158>
  - [46] R. Roth, "Maximum-rank array codes and their application to crisscross error correction," *IEEE Trans. Inf. Theory*, vol. 37, no. 2, pp. 328–336, Mar. 1991.
  - [47] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, ser. North-Holland Mathematical Library. Elsevier, 1977. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S092465090870521X>
  - [48] N. Shah, K. Rashmi, P. Kumar, and K. Ramchandran, "Interference alignment in regenerating codes for distributed storage: Necessity and code constructions," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2134–2158, Apr. 2012.
  - [49] C. Suh and K. Ramchandran, "Exact-repair MDS codes for distributed storage using interference alignment," in *Proc. 2010 IEEE International Symposium on Information Theory Proceedings (ISIT 2010)*, Jun. 2010, pp. 161–165.
  - [50] E. M. Gabidulin, A. V. Paramonov, and O. V. Tretjakov, "Ideals over a non-commutative ring and their application in cryptology," in *Advances in Cryptology - EUROCRYPT '91*, ser. Lecture Notes in Computer Science, vol. 547. Berlin, Heidelberg: Springer-Verlag, 1991, pp. 482–489. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1754868.1754922>
  - [51] K. Gibson, "The security of the Gabidulin public key cryptosystem," in *Advances in Cryptology - EUROCRYPT '96*, ser. Lecture Notes in Computer Science, vol. 1070. Springer, 1996, pp. 212–223.
  - [52] R. Koetter and F. R. Kschischang, "Coding for errors and erasures in random network coding," in *Proc. 2007 IEEE International Symposium on Information Theory (ISIT 2007)*, Jun. 2007.
  - [53] D. Silva, F. R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 3951–3967, Sep. 2008.